



ИНТЕЛЛЕКТУАЛЬНЫЙ
МЕГАПОЛИС

ЗАДАЧНИК



Инженерный класс

В МОСКОВСКОЙ ШКОЛЕ

НАПРАВЛЕНИЕ
КОСМИЧЕСКИЕ КЛАССЫ

ТЕОРЕТИЧЕСКИЙ ЭТАП



МОСКВА
2024



Оглавление

Вариант 1	3
Вариант 2	11
Вариант 3	19
Вариант 4	27
Вариант 5	35
Вариант 6	43
Вариант 7	51
Вариант 8	59
Вариант 9	67
Вариант 10	75
Вариант 11	83
Вариант 12	91
Вариант 13	99
Вариант 14	107
Вариант 15	115
Вариант 16	123

Вариант 1

Задание 1

Жители планеты Тинь для того, чтобы избежать сдвига привычных времён года, ввели понятие високосного года (подобно тому, как это сделано в юлианском или григорианском земных календарях). В обычном Тиньском году ровно 145 Тиньских дней. Високосным является каждый 4-й год, а также каждый 5-й год. В такие годы добавляется один дополнительный день. Если же очередной 4-й год совпадает с очередным 5-м, то добавляется только один день. Найдите точную продолжительность года на планете Тинь (выраженную в Тиньских днях). Выберите из предложенных вариантов ответа, на сколько точная продолжительность года отличается от 145 дней.

Варианты ответа:

- А) 1/2
- Б) 2/5
- В) 5/9
- Г) 4/29

Ответ: Б.

Задание 2

Космическое агентство разрабатывает автономного робота для исследования далёких планет. Робот должен уметь анализировать данные и выполнять повторяющиеся задачи в зависимости от полученных данных. Для этого разработана программа на языке C++, в которой используется несколько видов циклов и работа с массивами данных.

Входные данные:

- Робот получает данные о температуре на различных участках маршрута по поверхности планеты. Данные представлены в виде одномерного массива целых чисел (*temperatures[]*), где каждый элемент массива – это температура на конкретном участке.
- Робот фиксирует наличие препятствий на каждом участке, что представлено в виде другого массива (*obstacles[]*), где 1 означает наличие препятствия, а 0 – его отсутствие.

Роботу необходимо проанализировать данные и выполнить следующие задачи:

- 1) Робот должен двигаться по поверхности планеты, пока температура не опустится ниже -40°C или пока не встретит большое препятствие (обозначенное в массиве как 1).
Определить, сколько участков робот пройдёт до остановки (переменная *count_before_stop*).
Участок, на котором робот остановился, считать не нужно.
- 2) Робот должен проводить фотосъёмку каждого участка, где температура выше 0°C, и определить, сколько снимков сделает робот до остановки (переменная *photo_count*). На участке, где остановился робот, можно провести фотосъёмку.
- 3) Робот должен подсчитать общее количество участков с нормальной температурой (от -40°C до +40°C) и без препятствий (переменная *count_normal_temp*).

Просуммируйте три числа, получившихся в результате выполнения указанных задач, и выберите из предложенных ниже вариантов ответа, чему равна эта сумма:

- А) 10
- Б) 12
- В) 13
- Г) 11

Код программы:

```
#include <iostream>
```

```
using namespace std;
```

```
int main() {
```

```
    int temperatures[] = {-25, -40, 40, 0, -45, 30, -20};
```

```
    int obstacles[] = {0, 0, 0, 0, 0, 1, 0};
```

```
    int n = sizeof(temperatures) / sizeof(temperatures[0]);
```

```
    int i = 0;
```

```
    while (i < n && temperatures[i] >= -40 && obstacles[i] == 0) {
```

```
        i++;
```

```
    }
```

```
    int count_before_stop = i;
```

```
    int j = 0;
```

```
    int photo_count = 0;
```

```
    do {
```

```
        if (temperatures[j] > 0) {
```

```
            photo_count++;
```

```
        }
```

```
        j++;
```

```
    } while (j <= count_before_stop);
```

```
    int count_normal_temp = 0;
```

```
    for (int k = 0; k < n; k++) {
```

```
        if (temperatures[k] >= -40 && temperatures[k] <= 40 && obstacles[k] == 0) {
```

```
            count_normal_temp++;
```

```

}

}

cout << "Количество участков до остановки: " << count_before_stop << endl;

cout << "Количество снимков: " << photo_count << endl;

cout << "Количество участков с нормальной температурой и без препятствий: " <<
count_normal_temp << endl;

return 0;

}

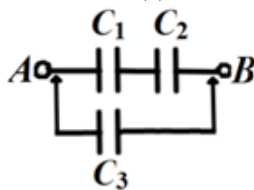
```

Ответ: А) 10.

Задание 3

При проведении серии опытов на борту космической станции батарею конденсаторов C_1 и C_2 зарядили до разности потенциалов 9,0 В на клеммах **A-B** (см. рисунок).

Затем источник отключили и к клеммам **A-B** подключили незаряженный плоский



конденсатор C_3 . Площадь его обкладок $3,61 \cdot 10^{-2} \text{ м}^2$, расстояние между ними $8,0 \cdot 10^{-6} \text{ м}$, диэлектрическая проницаемость вещества, заполняющего пространство между обкладками, $\epsilon = 40$. Ёмкости конденсаторов: $C_1 = 2,0 \text{ мкФ}$, $C_2 = 8,0 \text{ мкФ}$. Рассчитайте максимальную энергию конденсатора C_3 .

Приставка $1 \text{ мк} = 10^{-6}$; $\epsilon_0 = 8,85 \cdot 10^{-12} \text{ Ф/м}$.

Варианты ответа:

- А) 1,6 мкДж
- Б) 14,4 мкДж
- В) 4,5 мкДж
- Г) 7,2 мкДж
- Д) **16,2 мкДж**

Ответ: Д) 16,2 мкДж

Задание 4

На поверхности неизвестной планеты вниз по наклонной плоскости под углом 30° к горизонту из неподвижного состояния начинает скользить брусок. Коэффициент трения бруска о плоскость 0,22. Через 0,49 м пути он приобрёл скорость 1,0 м/с. Считая форму

планеты шаром радиуса 90 км и учитывая результаты эксперимента, определите массу планеты. Результат представьте в виде числа « $M \cdot 10^{20}$ кг» и выберите ответ с мантиссой «М».

Гравитационная постоянная $G = 6,67 \cdot 10^{-11} \text{ Н} \cdot \text{м}^2/\text{кг}^2$.

Возможные варианты ответа:

- А) 1
- Б) 2
- В) 3
- Г) 4**
- Д) 5
- Е) 6
- Ж) 7

Ответ: Г) 4

Задание 5

Некоторая планета совершает полный оборот вокруг своей звезды за 400 часов, а сутки (для наблюдателя на этой планете) длятся 50 часов. За сколько часов эта планета совершает один оборот вокруг своей оси, если направление вращения вокруг оси и вокруг звезды совпадают? Выберите один вариант ответа из списка ниже с учётом округления.

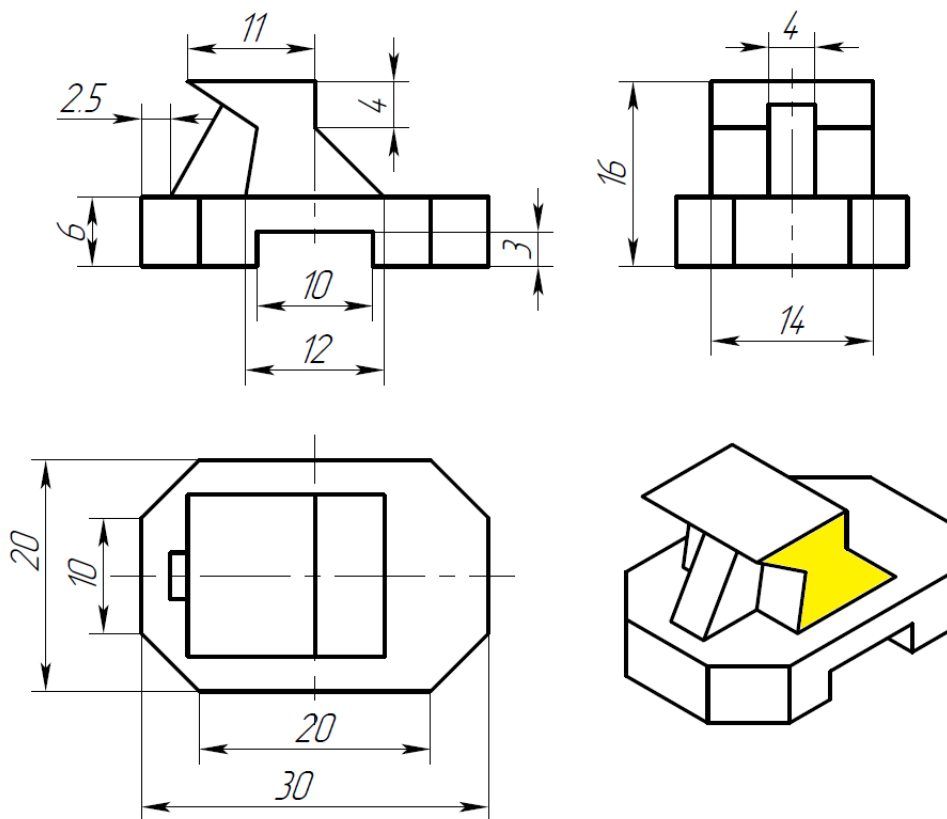
Варианты ответа:

- А) 44,44**
- Б) 50,00
- В) 57,14
- Г) 47,06

Ответ: А) 44,44.

Задание 6

По размерам, имеющимся на чертеже, определите площадь выделенной цветом грани.



Варианты ответа:

- А) 73
- Б) 83**
- В) 87
- Г) 90

Ответ: Б) 83

Задание 7

Найдите первую и вторую космические скорости на планете с плотностью $8,6 \text{ г/см}^3$ при её радиусе 8300 км. При расчётах принять гравитационную постоянную равной $6,67 \cdot 10^{-11} \text{ Н} \cdot \text{м}^2/\text{кг}^2$. Ответ выразите в км/с с одним знаком после запятой. При расчёте второй космической скорости необходимо воспользоваться полученным значением первой. В поля ответа необходимо ввести только числа.

Ответ: Первая космическая скорость: 12,9 км/с.

Вторая космическая скорость: 18,2 км/с.

Задание 8

Два тела движутся в одной плоскости. Если ввести на этой плоскости декартову систему координат, то траектории их движения совпадут с графиками функций $f(x) = 1 - x^2/12$ и $g(x) = \sin(\pi x - \pi/3)$. Определите количество точек пересечения этих траекторий.

Ответ: 10.

Задание 9

Космический крейсер оказался под атакой космических пиратов и отправил сигнал SOS. Два отряда космодесантников, находящихся на базе, получили сигнал и направились на помощь.

От базы (узел А) до крейсера (узел Д) можно добраться через сеть червоточин, соединяющих некоторые узлы в пространстве (планеты или точки в пространстве).

Космическое пространство можно представить в виде графа с пятью вершинами, где вершины – это узлы (планеты или точки в пространстве), а рёбра – это червоточины с определённой длиной. Ниже в матрице смежности дана информация о длине рёбер-червоточин.

	А	Б	В	Г	Д
А	0	4	8	3	0
Б	4	0	2	3	9
В	8	2	0	4	3
Г	3	3	4	0	9
Д	0	9	3	9	0

Отряд Альфа может перемещаться только по рёбрам с нечётной длиной.

Отряд Бета обязан чередовать движение по рёбрам с чётной и нечётной длиной, при этом чётность длины первого ребра не важна.

1) Какой из отрядов доберётся до крейсера быстрее? В случае равенства длин маршрутов выберите тот отряд, который прошёл через меньшее число узлов. В поле ответа введите название отряда одним словом с большой буквы без дополнительных символов (Альфа или Бета). **Ответ:** Бета.

2) Возможно ли, что один из отрядов не сможет добраться до крейсера, учитывая ограничения на рёбра? Напишите в поле ответа знак «+», если это возможно, или знак «-» –

в противном случае (без пробелов, кавычек или иных посторонних символов). **Ответ:** -.

3) Введите в поле ответа длину кратчайшего маршрута (одним числом без пробелов и дополнительных знаков). **Ответ:** 10.

Задание 10

Космическое агентство разработало программу на языке C++ для управления жизнеобеспечением и расчёта траектории корабля. Программа содержит три функции, работающие с локальными и глобальными переменными, а также использующими рекурсию.

Функция `calculateTrajectory` принимает расстояние до цели (в световых годах) и возвращает координаты корабля после полёта. Функция работает с глобальной переменной `currentPosition`, которая обновляется на основе пройденного расстояния.

Функция `adjustOxygenLevel` принимает количество дней полёта и суточное потребление кислорода на одного члена экипажа. Функция уменьшает глобальный уровень кислорода и выводит предупреждение, если его остаток ниже 10%.

Рекурсивная функция `calculateSurvivalDays` принимает начальный запас пищи, количество членов экипажа и суточное потребление пищи на одного члена экипажа и возвращает количество дней, на которые хватит запасов пищи.

Входные данные:

- Начальная позиция корабля (переменная `currentPosition`): 0 световых лет.
- Начальный запас кислорода (переменная `oxygenLevel`): 500 литров.
- Суточное потребление кислорода на одного члена экипажа (переменная `dailyOxygenConsumption`): 2 литра.
- Количество членов экипажа (переменная `crewMembers`): 4 человека.
- Начальный запас пищи (переменная `foodLevel`): 300 килограммов.
- Суточное потребление пищи на одного члена экипажа (переменная `dailyFoodConsumption`): 1.5 килограмма.
- Расстояние до цели (переменная `targetDistance`): 10 световых лет.
- Количество дней полёта (переменная `flightDays`): 20 дней.

Вопросы:

1. Как изменятся координаты корабля после полёта на 10 световых лет? **Ответ:** 10.
2. Какой будет уровень кислорода после 20 дней полёта? **Ответ:** 340.
3. На сколько дней хватит пищи для экипажа? **Ответ:** 50.

Код программы:

```
#include <iostream>
using namespace std;
```

```

double currentPosition;
double oxygenLevel;
double foodLevel;

double calculateTrajectory(double distance) {
    currentPosition += distance;
    return currentPosition;
}

void adjustOxygenLevel(int days, double dailyOxygenConsumption, int crewMembers) {
    oxygenLevel -= days * dailyOxygenConsumption * crewMembers;
    if (oxygenLevel < 0.1 * 500) {
        cout << "Внимание: Уровень кислорода ниже 10%!" << endl;
    }
}

int calculateSurvivalDays(double food, int crewMembers, double dailyFoodConsumption) {
    if (food < crewMembers * dailyFoodConsumption) {
        return 0;
    }
    return 1 + calculateSurvivalDays(food - crewMembers * dailyFoodConsumption,
    crewMembers, dailyFoodConsumption);
}

int main() {
    currentPosition = 0;
    oxygenLevel = 500;
    foodLevel = 300;

    double targetDistance = 10;
    int flightDays = 20;
    double dailyOxygenConsumption = 2;
    int crewMembers = 4;
    double dailyFoodConsumption = 1.5;

    cout << "Координаты после полёта: " << calculateTrajectory(targetDistance) << "
    световых лет" << endl;
    adjustOxygenLevel(flightDays, dailyOxygenConsumption, crewMembers);
    cout << "Уровень кислорода после " << flightDays << " дней: " << oxygenLevel << "
    литров" << endl;
    int survivalDays = calculateSurvivalDays(foodLevel, crewMembers, dailyFoodConsumption);
    cout << "Запаса пищи хватит на " << survivalDays << " дней" << endl;

    return 0;
}

```

Вариант 2

Задание 1

Жители планеты Тинь для того, чтобы избежать сдвига привычных времён года, ввели понятие високосного года (подобно тому, как это сделано в юлианском или григорианском земных календарях). В обычном Тиньском году ровно 54 Тиньских дня. Високосным является каждый 5-й год, а также каждый 9-й год. В такие годы добавляется один дополнительный день. Если же очередной 5-й год совпадает с очередным 9-м, то добавляется только один день. Найдите точную продолжительность года на планете Тинь (выраженную в Тиньских днях). Выберите из предложенных вариантов ответа, на сколько точная продолжительность года отличается от 54 дней.

Варианты ответа:

- А) 1/3
- Б) 5/9
- В) 13/45**
- Г) 45/54

Ответ: В) 13/45

.

Задание 2

Космическое агентство разрабатывает автономного робота для исследования далёких планет. Робот должен уметь анализировать данные и выполнять повторяющиеся задачи в зависимости от полученных данных. Для этого разработана программа на языке C++, в которой используется несколько видов циклов и работа с массивами данных.

Входные данные:

- Робот получает данные о температуре на различных участках маршрута по поверхности планеты. Данные представлены в виде одномерного массива целых чисел (*temperatures[]*), где каждый элемент массива – это температура на конкретном участке.
- Робот фиксирует наличие препятствий на каждом участке, что представлено в виде другого массива (*obstacles[]*), где 1 означает наличие препятствия, а 0 – его отсутствие.

Роботу необходимо проанализировать данные и выполнить следующие задачи:

- 1) Робот должен двигаться по поверхности планеты, пока температура не опустится ниже -40°C или пока не встретит большое препятствие (обозначенное в массиве как 1).
Определить, сколько участков робот пройдёт до остановки (переменная *count_before_stop*).
Участок, на котором робот остановился, считать не нужно.
- 2) Робот должен проводить фотосъёмку каждого участка, где температура ниже 0°C , и определить, сколько снимков сделает робот до остановки (переменная *photo_count*). На участке, где остановился робот, можно провести фотосъёмку.
- 3) Робот должен подсчитать общее количество участков с нормальной температурой (от -40°C до $+40^{\circ}\text{C}$) и без препятствий (переменная *count_normal_temp*).

Просуммируйте три числа, получившихся в результате выполнения указанных задач, и выберите из предложенных ниже вариантов ответа, чему равна эта сумма:

- A) 10
- Б) 12**
- В) 13
- Г) 11

Код программы:

```
#include <iostream>
```

```
using namespace std;
```

```
int main() {
```

```
    int temperatures[] = {-25, -40, 40, 0, -45, 30, -20};
```

```
    int obstacles[] = {0, 0, 0, 0, 0, 1, 0};
```

```
    int n = sizeof(temperatures) / sizeof(temperatures[0]);
```

```
    int i = 0;
```

```
    while (i < n && temperatures[i] >= -40 && obstacles[i] == 0) {
```

```
        i++;
```

```
    }
```

```
    int count_before_stop = i;
```

```
    int j = 0;
```

```
    int photo_count = 0;
```

```
    do {
```

```
        if (temperatures[j] < 0) {
```

```
            photo_count++;
```

```
        }
```

```
        j++;
```

```
    } while (j <= count_before_stop);
```

```
    int count_normal_temp = 0;
```

```
    for (int k = 0; k < n; k++) {
```

```
        if (temperatures[k] >= -40 && temperatures[k] <= 40 && obstacles[k] == 0) {
```

```
            count_normal_temp++;
```

```

}

}

cout << "Количество участков до остановки: " << count_before_stop << endl;

cout << "Количество снимков: " << photo_count << endl;

cout << "Количество участков с нормальной температурой и без препятствий: " <<
count_normal_temp << endl;

return 0;

}

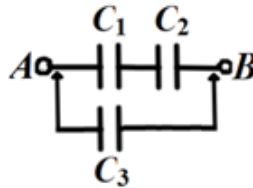
```

Ответ: Б) 12.

Задание 3

При проведении серии опытов на борту космической станции батарею конденсаторов C_1 и C_2 зарядили, подключив к клеммам **A-B** источник Э.Д.С. (см. рисунок).

После отключения источника к клеммам **A-B** подключили незаряженный конденсатор C_3 .



После максимальной зарядки конденсатора C_3 энергия конденсатора C_2 стала равна 16,2 мкДж. Ёмкости конденсаторов: $C_1 = 3,0$ мкФ, $C_2 = 2,0$ мкФ. Площадь обкладок плоского конденсатора C_3 равна $6,5 \cdot 10^{-2}$ м², расстояние между ними $8,0 \cdot 10^{-5}$ м, диэлектрическая проницаемость вещества, заполняющего пространство между обкладками $\epsilon = 57$. Вычислите максимальный заряд конденсатора C_3 .

Приставка 1мк = 10^{-6} ; $\epsilon_0 = 8,85 \cdot 10^{-12}$ Ф/м.

Варианты ответа:

- А) 0,41 мкКл
- Б) 1,20 мкКл
- В) 6,71 мкКл
- Г) **2,75 мкКл**
- Д) 16,2 мкКл

Ответ: Г) 2,75 мкКл.

Задание 4

На поверхности неизвестной планеты вверх по наклонной плоскости под углом 30° к горизонту толкнули брусок с начальной скоростью 10 м/с . Коэффициент трения бруска о плоскость $0,22$. Через $14,5 \text{ м}$ пути брусок прекратил двигаться вверх. Считая форму планеты шаром радиуса $4,0 \cdot 10^6 \text{ м}$ и учитывая результаты эксперимента, определите массу планеты. Результат представьте в виде числа « $M \cdot 10^{24} \text{ кг}$ » и выберите ответ с мантиссой «М».

Гравитационная постоянная $G = 6,67 \cdot 10^{-11} \text{ Н} \cdot \text{м}^2/\text{кг}^2$.

Возможные варианты ответа:

- А) 1,2**
- Б) 2,3
- В) 3,4
- Г) 4,5
- Д) 5,6
- Е) 6,7
- Ж) 7,8

Ответ: А) 1,2.

Задание 5

Некоторая планета совершает полный оборот вокруг своей звезды за 576 часов, а сутки (для наблюдателя на этой планете) длятся 72 часа. За сколько часов эта планета совершает один оборот вокруг своей оси, если направление вращения вокруг оси и вокруг звезды совпадают? Выберите один вариант ответа из списка ниже с учётом округления.

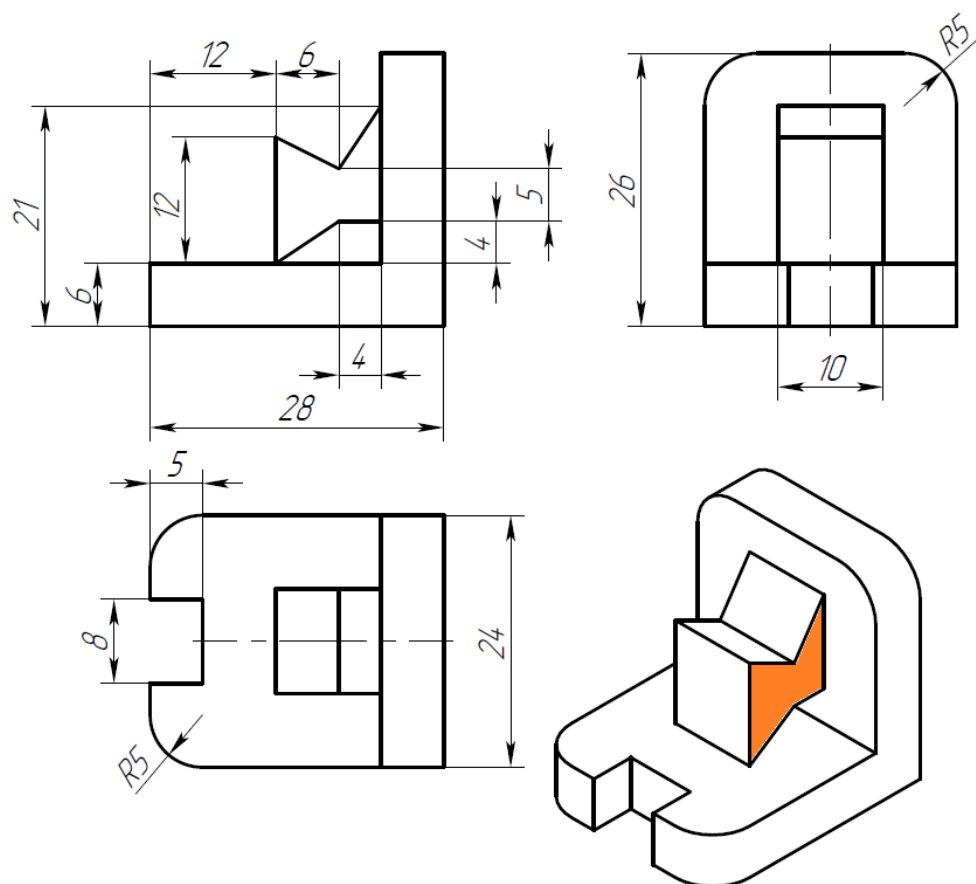
Варианты ответа:

- А) 72,00
- Б) 64,00**
- В) 57,60
- Г) 67,76

Ответ: Б) 64,00.

Задание 6

По размерам, имеющимся на чертеже, определите площадь выделенной цветом грани.



Варианты ответа:

- А) 73
- Б) 83**
- В) 87
- Г) 90

Ответ: Б) 83

Задание 7

Найдите первую и вторую космические скорости на планете с плотностью $5,3 \text{ г/см}^3$ при её радиусе 9200 км. При расчётах принять гравитационную постоянную равной $6,67 \cdot 10^{-11} \text{ Н} \cdot \text{м}^2/\text{кг}^2$. Ответ выразите в км/с с одним знаком после запятой. При расчёте второй космической скорости необходимо воспользоваться полученным значением первой. В поля ответа необходимо ввести только числа.

Ответ: Первая космическая скорость: 11,2 км/с.

Вторая космическая скорость: 15,8 км/с.

Задание 8

Два тела движутся в одной плоскости. Если ввести на этой плоскости декартову систему координат, то траектории их движения совпадут с графиками функций $f(x) = 1 - x^2/16$ и $g(x) = \sin(\pi x + \pi/4)$. Определите количество точек пересечения этих траекторий.

Ответ: 12.

Задание 9

Космический крейсер оказался под атакой космических пиратов и отправил сигнал SOS. Два отряда космодесантников, находящихся на базе, получили сигнал и направились на помощь.

От базы (узел А) до крейсера (узел Д) можно добраться через сеть червоточин, соединяющих некоторые узлы в пространстве (планеты или точки в пространстве).

Космическое пространство можно представить в виде графа с пятью вершинами, где вершины – это узлы (планеты или точки в пространстве), а рёбра – это червоточины с определённой длиной. Ниже в матрице смежности дана информация о длине рёбер-червоточин.

	А	Б	В	Г	Д
А	0	6	6	7	0
Б	6	0	3	2	8
В	6	3	0	1	9
Г	7	2	1	0	5
Д	0	8	9	5	0

Отряд Альфа может перемещаться только по рёбрам с нечётной длиной.

Отряд Бета обязан чередовать движение по рёбрам с чётной и нечётной длиной, при этом всегда начиная с чётной.

1) Какой из отрядов доберётся до крейсера быстрее? В случае равенства длин маршрутов выберите тот отряд, который прошёл через меньшее число узлов. В поле ответа введите название отряда одним словом с большой буквы без дополнительных символов (Альфа или Бета). **Ответ:** Альфа.

2) Возможно ли, что один из отрядов не сможет добраться до крейсера, учитывая

ограничения на рёбра? Напишите в поле ответа знак «+», если это возможно, или знак «-» – в противном случае (без пробелов, кавычек или иных посторонних символов). **Ответ:** -.

3) Введите в поле ответа длину кратчайшего маршрута (одним числом без пробелов и дополнительных знаков). **Ответ:** 12.

Задание 10

Космическое агентство разработало программу на языке C++ для управления энергией и управления полётом на исследовательском космическом корабле. Программа содержит три функции, работающие с локальными и глобальными переменными, а также использующими рекурсию.

Функция `calculateEnergyUsage` принимает количество дней полёта и ежедневное потребление энергии. Функция уменьшает глобальный уровень энергии и возвращает оставшийся запас.

Функция `adjustShieldLevel` принимает количество дней полёта и излучение в космосе. Она уменьшает уровень защиты от радиации и выводит предупреждение, если защита падает ниже 20%.

Рекурсивная функция `calculateMaxFlightTime` принимает запас топлива, ежедневное потребление топлива и возвращает максимальное количество дней, которое корабль может провести в полёте с имеющимся топливом.

Входные данные:

- Начальный уровень энергии (переменная `energyLevel`): 10000 единиц.
- Начальный уровень защиты от радиации (переменная `shieldLevel`): 500 единиц.
- Суточное потребление энергии (переменная `dailyEnergyConsumption`): 300 единиц.
- Уровень космического излучения (переменная `radiationLevel`): 5 единиц в день.
- Начальный запас топлива (переменная `fuelLevel`): 200 тонн.
- Суточное потребление топлива (переменная `dailyFuelConsumption`): 2 тонны.
- Количество дней полёта (переменная `flightDays`): 10 дней.

Вопросы:

1. Какой уровень энергии останется после 10 дней полёта? **Ответ:** 7000.
2. Как изменится уровень защиты после 10 дней полёта? **Ответ:** 450.
3. Сколько дней может продлиться полёт на оставшемся запасе топлива? **Ответ:** 100.

Код программы:

```
#include <iostream>
using namespace std;
```

```

double energyLevel;
double shieldLevel;
double fuelLevel;

double calculateEnergyUsage(int days, double dailyEnergyConsumption) {
    energyLevel -= days * dailyEnergyConsumption;
    return energyLevel;
}

void adjustShieldLevel(int days, double radiationLevel) {
    shieldLevel -= days * radiationLevel;
    if (shieldLevel < 0.2 * 500) {
        cout << "Внимание: Уровень защиты от радиации ниже 20%!" << endl;
    }
}

int calculateMaxFlightTime(double fuel, double dailyFuelConsumption) {
    if (fuel < dailyFuelConsumption) {
        return 0;
    }
    return 1 + calculateMaxFlightTime(fuel - dailyFuelConsumption, dailyFuelConsumption);
}

int main() {
    energyLevel = 10000;
    shieldLevel = 500;
    fuelLevel = 200;

    int flightDays = 10;
    double dailyEnergyConsumption = 300;
    double radiationLevel = 5;
    double dailyFuelConsumption = 2;

    cout << "Остаток энергии после " << flightDays << " дней: " <<
    calculateEnergyUsage(flightDays, dailyEnergyConsumption) << " единиц" << endl;
    adjustShieldLevel(flightDays, radiationLevel);
    cout << "Уровень защиты после " << flightDays << " дней: " << shieldLevel << "
    единиц" << endl;
    int maxFlightDays = calculateMaxFlightTime(fuelLevel, dailyFuelConsumption);
    cout << "Запас топлива хватит на " << maxFlightDays << " дней полёта" << endl;

    return 0;
}

```

Вариант 3

Задание 1

Жители планеты Тинь для того, чтобы избежать сдвига привычных времён года, ввели понятие високосного года (подобно тому, как это сделано в юлианском или григорианском земных календарях). В обычном Тиньском году ровно 200 Тиньских дней. Високосным является каждый 7-й год, а также каждый 8-й год. В такие годы добавляется один дополнительный день. Если же очередной 7-й год совпадает с очередным 8-м, то добавляется только один день. Найдите точную продолжительность года на планете Тинь (выраженную в Тиньских днях). Выберите из предложенных вариантов ответа, на сколько точная продолжительность года отличается от 200 дней.

Варианты ответа:

- А) 1/4
- Б) 7/15
- В) 15/56
- Г) 7/100

Ответ: А) 1/4.

Задание 2

Космическое агентство разрабатывает автономного робота для исследования далёких планет. Робот должен уметь анализировать данные и выполнять повторяющиеся задачи в зависимости от полученных данных. Для этого разработана программа на языке C++, в которой используется несколько видов циклов и работа с массивами данных.

Входные данные:

- Робот получает данные о температуре на различных участках маршрута по поверхности планеты. Данные представлены в виде одномерного массива целых чисел (*temperatures[]*), где каждый элемент массива – это температура на конкретном участке.
- Робот фиксирует наличие препятствий на каждом участке, что представлено в виде другого массива (*obstacles[]*), где 1 означает наличие препятствия, а 0 – его отсутствие.

Роботу необходимо проанализировать данные и выполнить следующие задачи:

- 1) Робот должен двигаться по поверхности планеты, пока температура не опустится ниже -40°C или пока не встретит большое препятствие (обозначенное в массиве как 1). Определить, сколько участков робот пройдёт до остановки (переменная *count_before_stop*). Участок, на котором робот остановился, считать не нужно.
- 2) Робот должен проводить фотосъёмку каждого участка, где температура не выше 0°C, и определить, сколько снимков сделает робот до остановки (переменная *photo_count*). На участке, где остановился робот, можно провести фотосъёмку.
- 3) Робот должен подсчитать общее количество участков с нормальной температурой (от -40°C до +40°C) и без препятствий (переменная *count_normal_temp*).

Просуммируйте три числа, получившихся в результате выполнения указанных задач, и выберите из предложенных ниже вариантов ответа, чему равна эта сумма:

- A) 10
- Б) 12
- В) 13**
- Г) 11

Код программы:

```
#include <iostream>
```

```
using namespace std;
```

```
int main() {
```

```
    int temperatures[] = {-25, -40, 40, 0, -45, 30, -20};
```

```
    int obstacles[] = {0, 0, 0, 0, 0, 1, 0};
```

```
    int n = sizeof(temperatures) / sizeof(temperatures[0]);
```

```
    int i = 0;
```

```
    while (i < n && temperatures[i] >= -40 && obstacles[i] == 0) {
```

```
        i++;
```

```
    }
```

```
    int count_before_stop = i;
```

```
    int j = 0;
```

```
    int photo_count = 0;
```

```
    do {
```

```
        if (temperatures[j] <= 0) {
```

```
            photo_count++;
```

```
        }
```

```
        j++;
```

```
    } while (j <= count_before_stop);
```

```
    int count_normal_temp = 0;
```

```
    for (int k = 0; k < n; k++) {
```

```
        if (temperatures[k] >= -40 && temperatures[k] <= 40 && obstacles[k] == 0) {
```

```
            count_normal_temp++;
```

```
        }
```

```

}

cout << "Количество участков до остановки: " << count_before_stop << endl;

cout << "Количество снимков: " << photo_count << endl;

cout << "Количество участков с нормальной температурой и без препятствий: " <<
count_normal_temp << endl;

return 0;

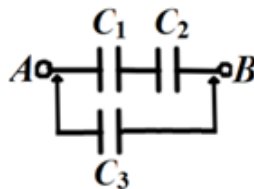
}

```

Ответ: В) 13.

Задание 3

При проведении серии опытов на борту космической станции батарее незаряженных конденсаторов C_1 и C_2 сообщили заряд $9 \cdot 10^{-4}$ Кл, подключив к клеммам **A-B** источник Э.Д.С. (см. рисунок).



Когда источник отключили, к клеммам **A-B** подключили незаряженный плоский конденсатор C_3 . Площадь его обкладок $3,4 \cdot 10^{-2} \text{ м}^2$, расстояние между ними $3,54 \cdot 10^{-7} \text{ м}$, диэлектрическая проницаемость вещества, заполняющего пространство между обкладками, $\epsilon = 40$. Ёмкости конденсаторов $C_1 = 280,0 \text{ мкФ}$, $C_2 = 70,0 \text{ мкФ}$. Рассчитайте энергию конденсатора C_1 после максимальной зарядки конденсатора C_3 .

Приставка $1 \text{ мк} = 10^{-6}$; $\epsilon_0 = 8,85 \cdot 10^{-12} \text{ Ф/м}$.

Варианты ответа:

- А) 34 мкДж
- Б) 900 мкДж
- В) 560 мкДж**
- Г) 320 мкДж
- Д) 160 мкДж

Ответ: В) 560 мкДж

Задание 4

На поверхности неизвестной планеты вниз по наклонной плоскости под углом 30° к горизонту из состояния покоя начинает скользить брусок. Коэффициент трения бруска о плоскость 0,22. За первые 0,9 с брусок преодолел 1 м пути. Считая форму планеты шаром,

её массу равной $3 \cdot 10^{24}$ кг и учитывая результаты эксперимента, определите радиус этой планеты. Результат представьте в виде числа «М·10⁶ м» и выберите ответ с мантиссой «М.

Гравитационная постоянная $G = 6,67 \cdot 10^{-11} \text{ Н} \cdot \text{м}^2/\text{кг}^2$.

Возможные варианты ответа:

- А) 1
- Б) 2
- В) 3
- Г) 4
- Д) 5**
- Е) 6
- Ж) 7

Ответ: Д) 5

Задание 5

Некоторая планета совершает полный оборот вокруг своей звезды за 720 часов, а сутки (для наблюдателя на этой планете) длятся 60 часов. За сколько часов эта планета совершает один оборот вокруг своей оси, если направление вращения вокруг оси и вокруг звезды совпадают? Выберите один вариант ответа из списка ниже с учётом округления.

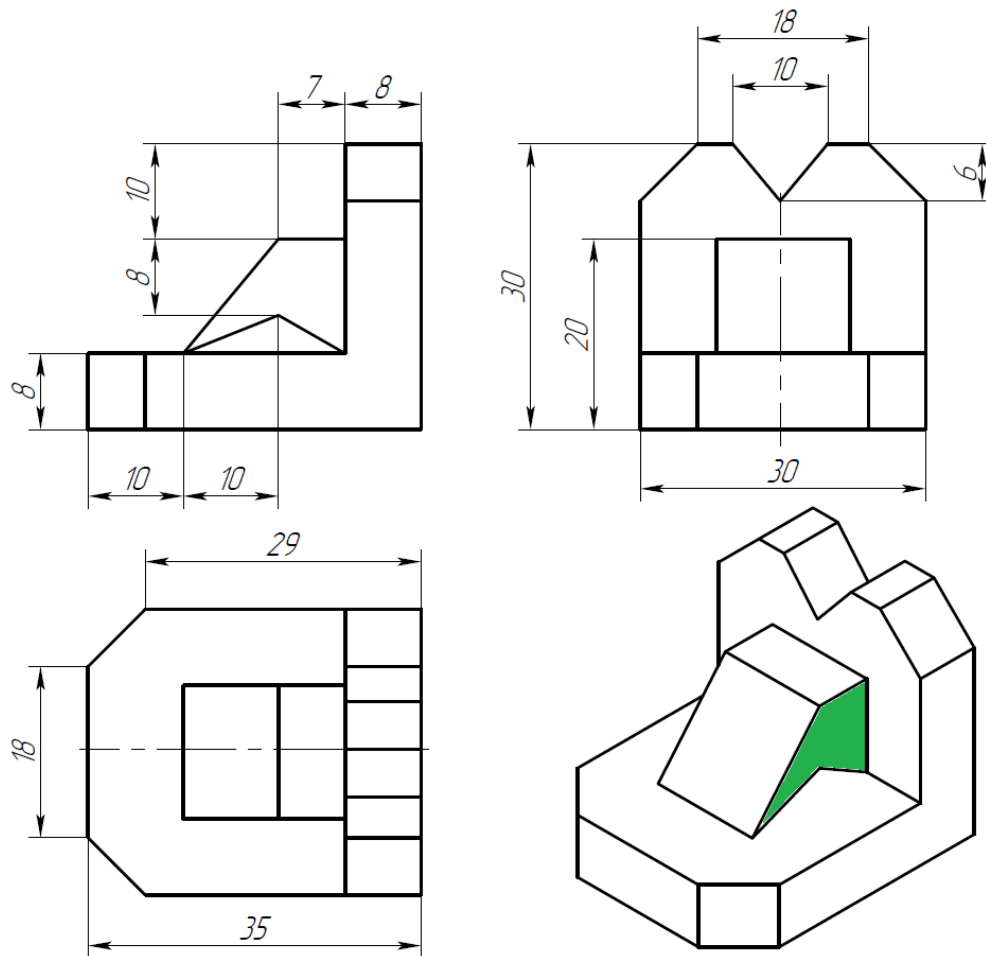
Варианты ответа:

- А) 55,38**
- Б) 65,45
- В) 60,00
- Г) 51,43

Ответ: А) 55,38.

Задание 6

По размерам, имеющимся на чертеже, определите площадь выделенной цветом грани.



Варианты ответа:

- А) 110
- Б) 100
- В) 120
- Г) 130

Ответ: А) 110

Задание 7

Найдите первую и вторую космические скорости на планете с плотностью $4,6 \text{ г/см}^3$ при её радиусе 5400 км. При расчётах принять гравитационную постоянную равной $6,67 \cdot 10^{-11} \text{ Н} \cdot \text{м}^2/\text{кг}^2$. Ответ выразите в км/с с одним знаком после запятой. При расчёте второй космической скорости необходимо воспользоваться полученным значением первой. В поля ответа необходимо ввести только числа.

Ответ: Первая космическая скорость: 61,2 км/с.

Вторая космическая скорость: 86,5 км/с.

Задание 8

Два тела движутся в одной плоскости. Если ввести на этой плоскости декартову систему координат, то траектории их движения совпадут с графиками функций $f(x) = 1 - \frac{(x-3)^2}{17}$ и $g(x) = \sin(\pi x)$. Определите количество точек пересечения этих траекторий.

Ответ: 12.

Задание 9

Космический крейсер оказался под атакой космических пиратов и отправил сигнал SOS. Два отряда космодесантников, находящихся на базе, получили сигнал и направились на помощь.

От базы (узел А) до крейсера (узел Д) можно добраться через сеть червоточин, соединяющих некоторые узлы в пространстве (планеты или точки в пространстве).

Космическое пространство можно представить в виде графа с пятью вершинами, где вершины – это узлы (планеты или точки в пространстве), а рёбра – это червоточины с определённой длиной. Ниже в матрице смежности дана информация о длине рёбер-червоточин.

	А	Б	В	Г	Д
А	0	7	2	6	0
Б	7	0	4	2	3
В	2	4	0	4	0
Г	6	2	4	0	4
Д	0	3	0	4	0

Отряд Альфа может перемещаться только по рёбрам с нечётной длиной.

Отряд Бета обязан чередовать движение по рёбрам с чётной и нечётной длиной, при этом четность длины первого ребра не важна.

1) Какой из отрядов доберётся до крейсера быстрее? В случае равенства длин маршрутов выберите тот отряд, который прошёл через меньшее число узлов. В поле ответа введите название отряда одним словом с большой буквы без дополнительных символов (Альфа или Бета). **Ответ:** Альфа.

2) Возможно ли, что один из отрядов не сможет добраться до крейсера, учитывая ограничения на рёбра? Напишите в поле ответа знак «+», если это возможно, или знак «-» – в противном случае (без пробелов, кавычек или иных посторонних символов). **Ответ:** +.

3) Введите в поле ответа длину кратчайшего маршрута (одним числом без пробелов и дополнительных знаков). **Ответ:** 10.

Задание 10

Космическое агентство разработало программу на языке C++ для управления запасами воды и поддержанием давления на борту исследовательского космического корабля. Программа использует три функции, работающие с локальными и глобальными переменными, а также реализующие рекурсию.

Функция `calculateWaterConsumption` принимает количество членов экипажа и количество дней полёта, возвращает общее количество воды, которое будет израсходовано за этот период.

Функция `adjustPressureControl` принимает текущее давление на борту и количество дней полёта. Она уменьшает или увеличивает глобальный уровень давления, чтобы поддерживать его в пределах нормы (90-110 единиц). Если давление выходит за эти пределы, выводится предупреждение.

Рекурсивная функция `calculateDaysUntilPressureFailure` принимает текущее давление, ежедневное падение давления, возвращает количество дней до того, как давление станет критическим.

Входные данные:

- Начальный уровень воды (переменная `waterLevel`): 1000 литров.
- Ежедневное потребление воды на одного члена экипажа (переменная `dailyWaterConsumption`): 3 литра.
- Количество членов экипажа (переменная `crewMembers`): 5 человек.
- Количество дней полёта (переменная `flightDays`): 15 дней.
- Начальный уровень давления (переменная `pressureLevel`): 120 единиц.
- Ежедневное падение давления (переменная `dailyPressureDrop`): 2 единицы.

Вопросы:

1. Сколько воды израсходует экипаж за 15 дней полёта? **Ответ:** 225.
2. Как изменится давление на борту после 15 дней полёта? **Ответ:** 110.
3. Через сколько дней давление достигнет критического уровня (менее 90 единиц)?

Ответ: 10.

Код программы:

```
#include <iostream>
using namespace std;
```

```
int waterLevel;
int pressureLevel;
```

```

int calculateWaterConsumption(int crewMembers, int days, int dailyWaterConsumption) {
    return crewMembers * days * dailyWaterConsumption;
}

void adjustPressureControl(int& pressure, int days) {
    for (int i = 0; i < days; i++) {
        if (pressure > 110) {
            pressure -= 1;
        } else if (pressure < 90) {
            pressure += 1;
        }
    }

    if (pressure > 110 || pressure < 90) {
        cout << "Внимание: Давление выходит за пределы нормы!" << endl;
    }
}

int calculateDaysUntilPressureFailure(int pressure, int dailyPressureDrop) {
    if (pressure <= 90) {
        return 0;
    }
    return 1 + calculateDaysUntilPressureFailure(pressure - dailyPressureDrop,
dailyPressureDrop);
}

int main() {
    waterLevel = 1000;
    pressureLevel = 120;
    int crewMembers = 5;
    int dailyWaterConsumption = 3;
    int flightDays = 15;
    int dailyPressureDrop = 2;

    int waterConsumed = calculateWaterConsumption(crewMembers, flightDays,
dailyWaterConsumption);
    cout << "Экипаж израсходует " << waterConsumed << " литров воды за " <<
flightDays << " дней полёта" << endl;

    adjustPressureControl(pressureLevel, flightDays);
    cout << "Давление на борту после " << flightDays << " дней: " << pressureLevel << "
единиц" << endl;

    int daysUntilPressureFailure = calculateDaysUntilPressureFailure(pressureLevel,
dailyPressureDrop);
    cout << "Давление достигнет критического уровня через " << daysUntilPressureFailure
<< " дней" << endl;

    return 0;
}

```

Вариант 4

Задание 1

Жители планеты Тинь для того, чтобы избежать сдвига привычных времён года, ввели понятие високосного года (подобно тому, как это сделано в юлианском или григорианском земных календарях). В обычном Тиньском году ровно 75 Тиньских дней. Високосным является каждый 3-й год, а также каждый 5-й год. В такие годы добавляется один дополнительный день. Если же очередной 3-й год совпадает с очередным 5-м, то добавляется только один день. Найдите точную продолжительность года на планете Тинь (выраженную в Тиньских днях). Выберите из предложенных вариантов ответа, на сколько точная продолжительность года отличается от 75 дней.

Варианты ответа:

- А) 3/5
- Б) 1/5
- В) 14/75
- Г) 7/15

Ответ: Г) 7/15.

Задание 2

Космическое агентство разрабатывает автономного робота для исследования далёких планет. Робот должен уметь анализировать данные и выполнять повторяющиеся задачи в зависимости от полученных данных. Для этого разработана программа на языке C++, в которой используется несколько видов циклов и работа с массивами данных.

Входные данные:

- Робот получает данные о температуре на различных участках маршрута по поверхности планеты. Данные представлены в виде одномерного массива целых чисел (*temperatures[]*), где каждый элемент массива – это температура на конкретном участке.
- Робот фиксирует наличие препятствий на каждом участке, что представлено в виде другого массива (*obstacles[]*), где 1 означает наличие препятствия, а 0 – его отсутствие.

Роботу необходимо проанализировать данные и выполнить следующие задачи:

- 1) Робот должен двигаться по поверхности планеты, пока температура не опустится ниже -40°C или пока не встретит большое препятствие (обозначенное в массиве как 1). Определить, сколько участков робот пройдёт до остановки (переменная *count_before_stop*). Участок, на котором робот остановился, считать не нужно.
- 2) Робот должен проводить фотосъёмку каждого участка, где температура не ниже 0°C, и определить, сколько снимков сделает робот до остановки (переменная *photo_count*). На участке, где остановился робот, можно провести фотосъёмку.
- 3) Робот должен подсчитать общее количество участков с нормальной температурой (от -40°C до +40°C) и без препятствий (переменная *count_normal_temp*).

Просуммируйте три числа, получившихся в результате выполнения указанных задач, и выберите из предложенных ниже вариантов ответа, чему равна эта сумма:

- A) 10
- Б) 12
- В) 13
- Г) 11

Код программы:

```
#include <iostream>
```

```
using namespace std;
```

```
int main() {
```

```
    int temperatures[] = {-25, -40, 40, 0, -45, 30, -20};
```

```
    int obstacles[] = {0, 0, 0, 0, 0, 1, 0};
```

```
    int n = sizeof(temperatures) / sizeof(temperatures[0]);
```

```
    int i = 0;
```

```
    while (i < n && temperatures[i] >= -40 && obstacles[i] == 0) {
```

```
        i++;
```

```
    }
```

```
    int count_before_stop = i;
```

```
    int j = 0;
```

```
    int photo_count = 0;
```

```
    do {
```

```
        if (temperatures[j] >= 0) {
```

```
            photo_count++;
```

```
        }
```

```
        j++;
```

```
    } while (j <= count_before_stop);
```

```
    int count_normal_temp = 0;
```

```
    for (int k = 0; k < n; k++) {
```

```
        if (temperatures[k] >= -40 && temperatures[k] <= 40 && obstacles[k] == 0) {
```

```
            count_normal_temp++;
```

```
        }
```

```

}

cout << "Количество участков до остановки: " << count_before_stop << endl;

cout << "Количество снимков: " << photo_count << endl;

cout << "Количество участков с нормальной температурой и без препятствий: " <<
count_normal_temp << endl;

return 0;

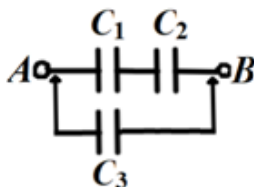
}

```

Ответ: Г) 11.

Задание 3

При проведении серии опытов на борту космической станции батарею конденсаторов C_1 и C_2 зарядили до суммарной энергии 264,6 мкДж, подключив к клеммам **A-B** источник Э.Д.С. (см. рисунок).



Затем источник отключили и к клеммам **A-B** подключили незаряженный плоский конденсатор C_3 . Площадь его обкладок $5,8 \cdot 10^{-2} \text{ м}^2$, расстояние между ними $6,0 \cdot 10^{-6} \text{ м}$, диэлектрическая проницаемость вещества, заполняющего пространство между обкладками, $\epsilon = 35$. Ёмкости конденсаторов: $C_1 = 1,6 \text{ мкФ}$, $C_2 = 4,8 \text{ мкФ}$. Рассчитайте разность потенциалов на обкладках конденсатора C_1 после максимальной зарядки конденсатора C_3 .

Приставка $1 \text{ мк} = 10^{-6}$; $\epsilon_0 = 8,85 \cdot 10^{-12} \text{ Ф/м}$.

Варианты ответа:

- А) 3,0 В
- Б) 4,5 В**
- В) 6,1 В
- Г) 0,9 В
- Д) 7,2 В

Ответ: Б) 4,5 В

Задание 4

На поверхности неизвестной планеты толкнули брусок вверх по наклонной плоскости. Угол наклона плоскости к горизонту составляет 30° , начальная скорость бруска 8 м/с.

Коэффициент трения бруска о плоскость 0,21. Через 19 м пути скорость бруска уменьшилась на 5 м/с. Считая форму планеты шаром, её массу равной $3 \cdot 10^{22}$ кг и учитывая результаты эксперимента, вычислите радиус планеты. Результат представьте в виде числа «М·10⁵ м» и выберите ответ с мантиссой «М».

Гравитационная постоянная $G = 6,67 \cdot 10^{-11} \text{ Н} \cdot \text{м}^2/\text{кг}^2$.

Возможные варианты ответа:

- А) 3
- Б) 4
- В) 5
- Г) 6
- Д) 7**
- Е) 8
- Ж) 9

Ответ: Д) 7.

Задание 5

Некоторая планета совершает полный оборот вокруг своей звезды за 624 часа, а сутки (для наблюдателя на этой планете) длятся 52 часа. За сколько часов эта планета совершает один оборот вокруг своей оси, если направление вращения вокруг оси и вокруг звезды совпадают? Выберите один вариант ответа из списка ниже с учётом округления.

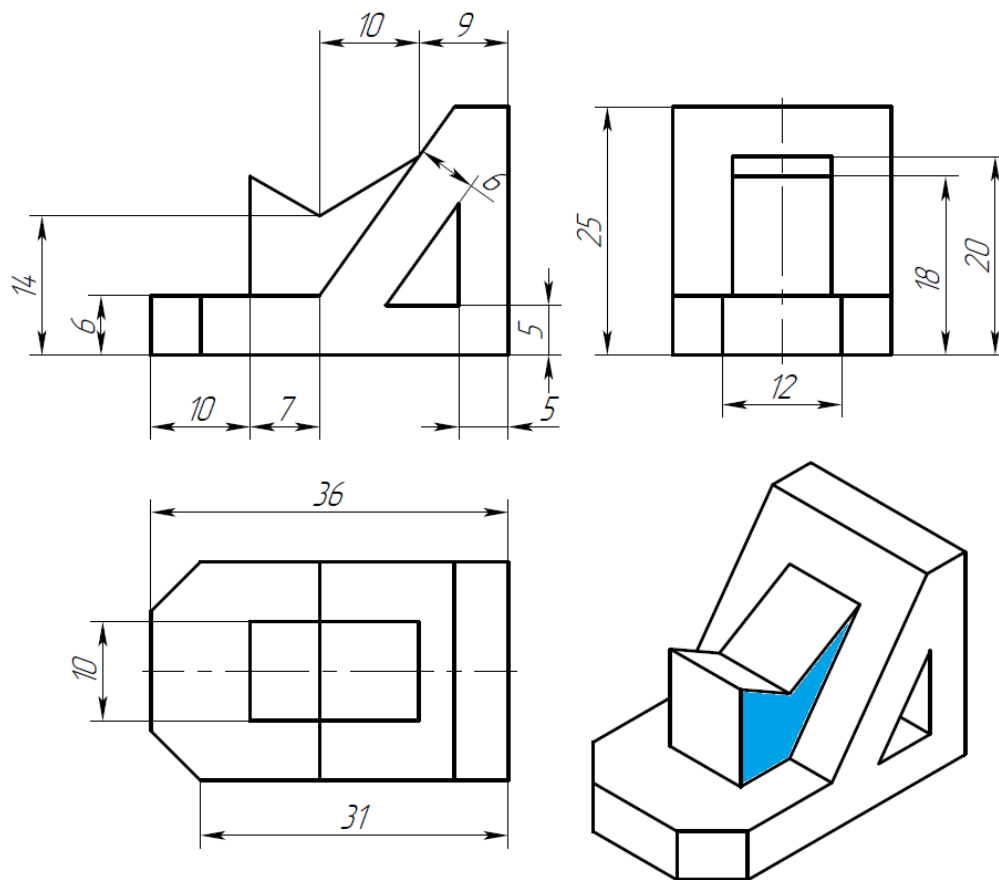
Варианты ответа:

- А) 44,57
- Б) 56,73
- В) 52,00
- Г) 48,00**

Ответ: Г) 48,00.

Задание 6

По размерам, имеющимся на чертеже, определите площадь выделенной цветом грани.



Варианты ответа:

- А) 100
- Б) 110**
- В) 120
- Г) 130

Ответ: Б) 110

Задание 7

Найдите первую и вторую космические скорости на планете с плотностью $6,8 \text{ г/см}^3$ при её радиусе 5600 км . При расчётах принять гравитационную постоянную равной $6,67 \cdot 10^{-11} \text{ Н} \cdot \text{м}^2/\text{кг}^2$. Ответ выразите в км/с с одним знаком после запятой. При расчёте второй космической скорости необходимо воспользоваться полученным значением первой. В поля ответа необходимо ввести только числа.

Ответ: Первая космическая скорость: 7,7 км/с.

Вторая космическая скорость: 10,9 км/с.

Задание 8

Два тела движутся в одной плоскости. Если ввести на этой плоскости декартову систему координат, то траектории их движения совпадут с графиками функций $f(x) = 1 - \frac{(x-1)^2}{8}$ и $g(x) = \sin(\pi x)$. Определите количество точек пересечения этих траекторий.

Ответ: 8.

Задание 9

Космический крейсер оказался под атакой космических пиратов и отправил сигнал SOS. Два отряда космодесантников, находящихся на базе, получили сигнал и направились на помощь.

От базы (узел А) до крейсера (узел Д) можно добраться через сеть червоточин, соединяющих некоторые узлы в пространстве (планеты или точки в пространстве).

Космическое пространство можно представить в виде графа с пятью вершинами, где вершины – это узлы (планеты или точки в пространстве), а рёбра – это червоточины с определённой длиной. Ниже в матрице смежности дана информация о длине рёбер-червоточин.

	А	Б	В	Г	Д
А	0	4	7	6	0
Б	4	0	3	2	8
В	7	3	0	1	2
Г	6	2	1	0	4
Д	0	8	2	4	0

Отряд Альфа может перемещаться только по рёбрам с нечётной длиной.

Отряд Бета обязан чередовать движение по рёбрам с чётной и нечётной длиной, при этом чётность длины первого ребра не важна.

1) Какой из отрядов доберётся до крейсера быстрее? В случае равенства длин маршрутов выберите тот отряд, который прошёл через меньшее число узлов. В поле ответа введите название отряда одним словом с большой буквы без дополнительных символов (Альфа или Бета). **Ответ:** Бета.

2) Возможно ли, что один из отрядов не сможет добраться до крейсера, учитывая

ограничения на рёбра? Напишите в поле ответа знак «+», если это возможно, или знак «-» – в противном случае (без пробелов, кавычек или иных посторонних символов). **Ответ:** +.

3) Введите в поле ответа длину кратчайшего маршрута (одним числом без пробелов и дополнительных знаков). **Ответ:** 9.

Задание 10

Космическое агентство разработало программу на языке C++ для управления запасом кислорода на борту исследовательского космического корабля. Программа использует три функции, работающие с локальными и глобальными переменными, а также реализующими рекурсию.

Функция `calculateOxygenConsumption` принимает количество членов экипажа и количество дней полёта, возвращает количество кислорода, которое экипаж израсходует.

Функция `checkOxygenLevel` принимает количество оставшегося кислорода и количество членов экипажа. Если уровень кислорода становится слишком низким (меньше 10% от начального запаса), выводится предупреждение.

Рекурсивная функция `calculateOxygenDaysLeft` принимает начальный запас кислорода и суточное потребление кислорода экипажем. Она возвращает количество дней, на которые хватит оставшегося кислорода.

Входные данные:

- Начальный запас кислорода (переменная `oxygenLevel`): 600 литров.
- Суточное потребление кислорода на одного члена экипажа (переменная `dailyOxygenConsumption`): 2 литра.
- Количество членов экипажа (переменная `crewMembers`): 4 человека.
- Количество дней полёта (переменная `flightDays`): 15 дней.

Вопросы:

1. Сколько кислорода израсходует экипаж за 15 дней полёта? **Ответ:** 120.
2. Какой будет уровень кислорода после 15 дней полёта? **Ответ:** 480.
3. На сколько дней хватит оставшегося кислорода? **Ответ:** 60.

Код программы:

```
#include <iostream>
using namespace std;

int oxygenLevel;

int calculateOxygenConsumption(int crewMembers, int days) {
    int oxygenConsumed = crewMembers * days * 2;
    return oxygenConsumed;
}
```

```

void checkOxygenLevel(int oxygen, int crewMembers) {
    if (oxygen < 0.1 * 600) {
        cout << "Внимание: Уровень кислорода слишком низкий!" << endl;
    }
}

int calculateOxygenDaysLeft(int oxygen, int dailyOxygenConsumption) {
    if (oxygen < dailyOxygenConsumption * 4) {
        return 0;
    }
    return 1 + calculateOxygenDaysLeft(oxygen - dailyOxygenConsumption * 4,
dailyOxygenConsumption);
}

int main() {
    oxygenLevel = 600;
    int crewMembers = 4;
    int flightDays = 15;
    int dailyOxygenConsumption = 2;

    int oxygenConsumed = calculateOxygenConsumption(crewMembers, flightDays);
    cout << "Экипаж израсходует " << oxygenConsumed << " литров кислорода за " <<
flightDays << " дней" << endl;

    int remainingOxygen = oxygenLevel - oxygenConsumed;
    checkOxygenLevel(remainingOxygen, crewMembers);
    cout << "Уровень кислорода после " << flightDays << " дней: " << remainingOxygen <<
" литров" << endl;

    int oxygenDaysLeft = calculateOxygenDaysLeft(remainingOxygen,
dailyOxygenConsumption);
    cout << "Запаса кислорода хватит на " << oxygenDaysLeft << " дней" << endl;

    return 0;
}

```

Вариант 5

Задание 1

Жители планеты Кор для того, чтобы избежать сдвига привычных времён года, ввели понятие високосного года (подобно тому, как это сделано в юлианском или григорианском земных календарях). В обычном Корском году ровно 150 Корских дней. Високосным является каждый 5-й год, а также каждый 11-й год. В такие годы добавляется один дополнительный день. Если же очередной 5-й год совпадает с очередным 11-м, то год считается обычным. Найдите точную продолжительность года на планете Кор (выраженную в Корских днях). Выберите из предложенных вариантов ответа, на сколько точная продолжительность года отличается от 150 дней.

Варианты ответа:

- А) 14/55
- Б) 3/11
- В) 16/55
- Г) 8/75

Ответ: А) 14/55.

Задание 2

Космическое агентство разрабатывает автономного робота для исследования далёких планет. Робот должен уметь анализировать данные и выполнять повторяющиеся задачи в зависимости от полученных данных. Для этого разработана программа на языке C++, в которой используется несколько видов циклов и работа с массивами данных.

Входные данные:

- Робот получает данные о температуре на различных участках маршрута по поверхности планеты. Данные представлены в виде одномерного массива целых чисел (*temperatures[]*), где каждый элемент массива – это температура на конкретном участке.
- Робот фиксирует наличие препятствий на каждом участке, что представлено в виде другого массива (*obstacles[]*), где 1 означает наличие препятствия, а 0 – его отсутствие.

Роботу необходимо проанализировать данные и выполнить следующие задачи:

- 1) Робот должен двигаться по поверхности планеты, пока температура не опустится ниже -40°C или пока не встретит большое препятствие (обозначенное в массиве как 1). Определить, сколько участков робот пройдёт до остановки (переменная *count_before_stop*). Участок, на котором робот остановился, считать не нужно.
- 2) Робот должен проводить фотосъёмку каждого участка, где температура выше 0°C, и определить, сколько снимков сделает робот до остановки (переменная *photo_count*). На участке, где остановился робот, можно провести фотосъёмку.
- 3) Робот должен подсчитать общее количество участков с нормальной температурой (от -40°C до +40°C) и без препятствий (переменная *count_normal_temp*).

Просуммируйте три числа, получившихся в результате выполнения указанных задач, и выберите из предложенных ниже вариантов ответа, чему равна эта сумма:

- A) 8
- Б) 9
- В) 10
- Г) 7

Код программы:

```
#include <iostream>
```

```
using namespace std;
```

```
int main() {
```

```
    int temperatures[] = {-25, -40, 40, 0, -45, 30, -20};
```

```
    int obstacles[] = {0, 0, 0, 1, 0, 1, 0};
```

```
    int n = sizeof(temperatures) / sizeof(temperatures[0]);
```

```
    int i = 0;
```

```
    while (i < n && temperatures[i] >= -40 && obstacles[i] == 0) {
```

```
        i++;
```

```
    }
```

```
    int count_before_stop = i;
```

```
    int j = 0;
```

```
    int photo_count = 0;
```

```
    do {
```

```
        if (temperatures[j] > 0) {
```

```
            photo_count++;
```

```
        }
```

```
        j++;
```

```
    } while (j <= count_before_stop);
```

```
    int count_normal_temp = 0;
```

```
    for (int k = 0; k < n; k++) {
```

```
        if (temperatures[k] >= -40 && temperatures[k] <= 40 && obstacles[k] == 0) {
```

```
            count_normal_temp++;
```

```
        }
```

```

}

cout << "Количество участков до остановки: " << count_before_stop << endl;

cout << "Количество снимков: " << photo_count << endl;

cout << "Количество участков с нормальной температурой и без препятствий: " <<
count_normal_temp << endl;

return 0;

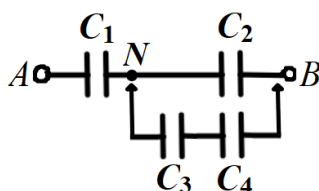
}

```

Ответ: А) 8.

Задание 3

При проведении серии опытов на борту космической станции батарею конденсаторов C_1 и C_2 зарядили до разности потенциалов 17,5 В на клеммах $A-B$ (см. рисунок).



Затем источник отключили и к клеммам $N-B$ подключили батарею незаряженных конденсаторов C_3 и C_4 . Площадь обкладок плоского конденсатора C_1 равна $8 \cdot 10^{-3} \text{ м}^2$, расстояние между ними $3,54 \cdot 10^{-6} \text{ м}$, диэлектрическая проницаемость вещества, заполняющего пространство между обкладками, $\epsilon = 50$. Ёмкости конденсаторов: $C_2 = 4,0 \text{ мкФ}$, $C_3 = C_4 = 6,0 \text{ мкФ}$. Рассчитайте энергию конденсатора C_2 после максимальной зарядки конденсаторов блока C_{34} .

Приставка $1 \text{ мк} = 10^{-6}$; $\epsilon_0 = 8,85 \cdot 10^{-12} \text{ Ф/м}$.

Варианты ответа:

- А) 6 мкДж
- Б) 7 мкДж
- В) 8 мкДж**
- Г) 9 мкДж
- Д) 10 мкДж

Ответ: В). 8 мкДж

Задание 4

На поверхности неизвестной планеты по горизонтальной плоскости толкнули брусок с начальной скоростью 4 м/с. Коэффициент трения бруска о плоскость 0,28. Через 5 м пути

скорость бруска уменьшилась на 3 м/с. Считая форму планеты шаром радиуса 190 км и учитывая результаты эксперимента, вычислите первую космическую скорость для этой планеты. Результат представьте в виде числа « $M \cdot 10^3$ м/с» и выберите ответ с мантиссой «М».

Гравитационная постоянная $G = 6,67 \cdot 10^{-11} \text{ Н} \cdot \text{м}^2/\text{кг}^2$.

Возможные варианты ответа:

- А) 1
- Б) 2
- В) 3
- Г) 4
- Д) 5
- Е) 6
- Ж) 7

Ответ: А) 1.

Задание 5

Некоторая планета совершает полный оборот вокруг своей звезды за 640 часов, а сутки (для наблюдателя на этой планете) длятся 40 часов. За сколько часов эта планета совершает один оборот вокруг своей оси, если направление вращения вокруг оси и вокруг звезды совпадают? Выберите один вариант ответа из списка ниже с учётом округления.

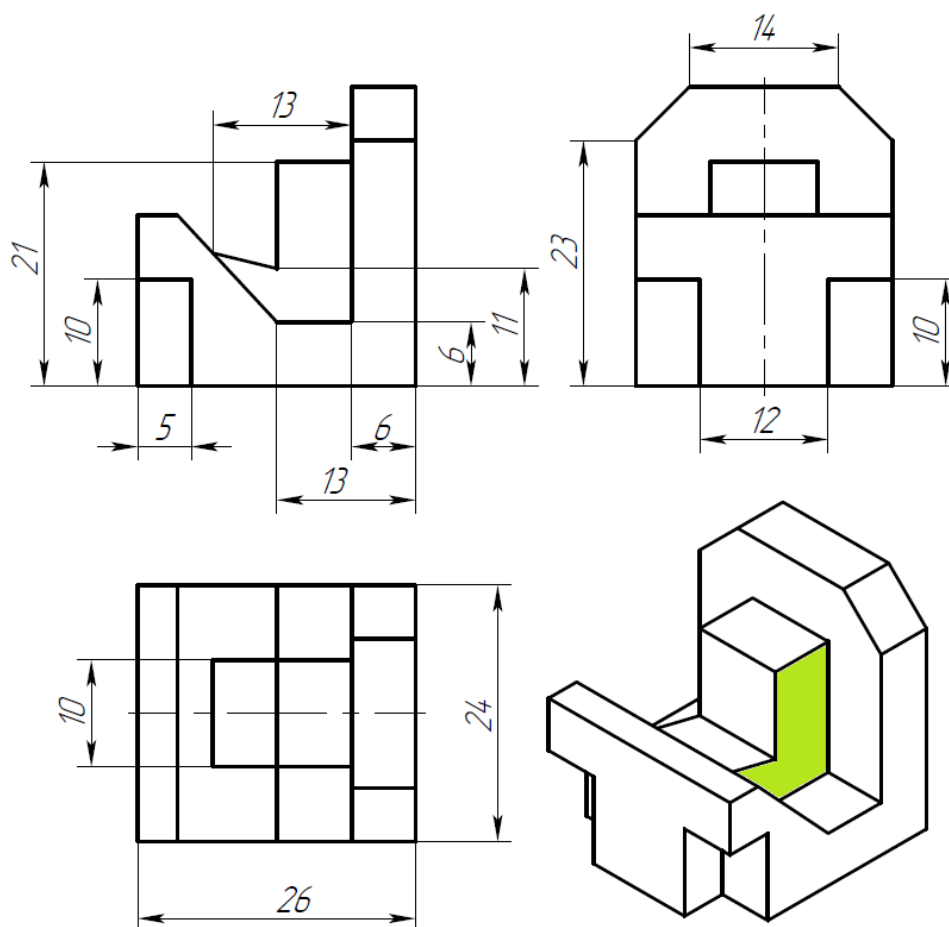
Варианты ответа:

- А) 42,67
- Б) 37,65**
- В) 35,56
- Г) 40,00

Ответ: Б) 37,65.

Задание 6

По размерам, имеющимся на чертеже, определите площадь выделенной цветом грани.



Варианты ответа:

- А) 100
- Б) 110
- В) 120**
- Г) 130

Ответ: В) 120

Задание 7

Найти первую и вторую космические скорости для планеты Нептун. Масса Нептуна составляет $1,04 \cdot 10^{26}$ кг, средний радиус – 24 622 км. При расчётах принять гравитационную постоянную равной $6,67 \cdot 10^{-11}$ Н·м²/кг². Ответ выразите в км/с с одним знаком после запятой. При расчёте второй космической скорости необходимо воспользоваться полученным значением первой. В поля ответа необходимо ввести только числа.

Ответ: Первая космическая скорость: 16,8 км/с.

Вторая космическая скорость: 23,8 км/с.

Задание 8

Два тела движутся в одной плоскости. Если ввести на этой плоскости декартову систему координат, то траектории их движения совпадут с графиками функций $f(x) = 1 - x^2/15$ и $g(x) = \cos(\pi x - 3\pi/4)$. Определите количество точек пересечения этих траекторий.

Ответ: 10.

Задание 9

Космический крейсер оказался под атакой космических пиратов и отправил сигнал SOS. Два отряда космодесантников, находящихся на базе, получили сигнал и направились на помощь.

От базы (узел А) до крейсера (узел Д) можно добраться через сеть червоточин, соединяющих некоторые узлы в пространстве (планеты или точки в пространстве).

Космическое пространство можно представить в виде графа с пятью вершинами, где вершины – это узлы (планеты или точки в пространстве), а рёбра – это червоточины с определённой длиной. Ниже в матрице смежности дана информация о длине рёбер-червоточин.

	А	Б	В	Г	Д
А	0	4	7	8	0
Б	4	0	6	2	8
В	7	6	0	4	0
Г	8	2	4	0	2
Д	0	8	0	2	0

Отряд Альфа обязан чередовать движение по рёбрам с чётной и нечётной длиной, при этом чётность длины первого ребра не важна.

Отряд Бета может перемещаться только по рёбрам с чётной длиной.

1) Какой из отрядов доберётся до крейсера быстрее? В случае равенства длин маршрутов выберите тот отряд, который прошёл через меньшее число узлов. В поле ответа введите название отряда одним словом с большой буквы без дополнительных символов (Альфа или Бета). **Ответ:** Бета.

2) Возможно ли, что один из отрядов не сможет добраться до крейсера, учитывая ограничения на рёбра? Напишите в поле ответа знак «+», если это возможно, или знак «-» –

в противном случае (без пробелов, кавычек или иных посторонних символов). **Ответ:** +.

3) Введите в поле ответа длину кратчайшего маршрута (одним числом без пробелов и дополнительных знаков). **Ответ:** 8.

Задание 10

Космическое агентство разработало программу на языке C++ для управления запасами пищи на борту исследовательского космического корабля. Программа использует три функции, работающие с локальными и глобальными переменными, а также реализующими рекурсию.

Функция `calculateFoodConsumption` принимает количество членов экипажа и количество дней полёта, возвращает количество еды, которую экипаж израсходует.

Функция `checkFoodLevel` принимает количество оставшейся пищи и количество членов экипажа. Если еды остаётся меньше 20% от начального запаса, выводится предупреждение.

Рекурсивная функция `calculateFoodDaysLeft` принимает оставшийся запас пищи и суточное потребление экипажем, возвращает количество дней, на которые хватит оставшегося запаса.

Входные данные:

- Начальный запас еды (переменная `foodLevel`): 500 кг.
- Суточное потребление пищи на одного члена экипажа (переменная `dailyFoodConsumption`): 3 кг.
- Количество членов экипажа (переменная `crewMembers`): 5 человек.
- Количество дней полёта (переменная `flightDays`): 10 дней.

Вопросы:

1. Сколько еды израсходует экипаж за 10 дней полёта? **Ответ:** 150.
2. Какой будет уровень оставшейся пищи после 10 дней полёта? **Ответ:** 350.
3. На сколько дней хватит оставшегося запаса еды? **Ответ:** 23.

Код программы:

```
#include <iostream>
using namespace std;

int foodLevel;

int calculateFoodConsumption(int crewMembers, int days) {
    int foodConsumed = crewMembers * days * 3;
    return foodConsumed;
}
```

```

void checkFoodLevel(int food, int crewMembers) {
    if (food < 0.2 * 500) {
        cout << "Внимание: Осталось меньше 20% еды!" << endl;
    }
}

int calculateFoodDaysLeft(int food, int dailyFoodConsumption) {
    if (food < dailyFoodConsumption * 5) {
        return 0;
    }
    return 1 + calculateFoodDaysLeft(food - dailyFoodConsumption * 5,
dailyFoodConsumption);
}

int main() {
    foodLevel = 500;
    int crewMembers = 5;
    int flightDays = 10;
    int dailyFoodConsumption = 3;

    int foodConsumed = calculateFoodConsumption(crewMembers, flightDays);
    cout << "Экипаж израсходует " << foodConsumed << " кг пищи за " << flightDays << "
дней" << endl;

    int remainingFood = foodLevel - foodConsumed;
    checkFoodLevel(remainingFood, crewMembers);
    cout << "Оставшийся запас еды после " << flightDays << " дней: " << remainingFood
<< " кг" << endl;

    int foodDaysLeft = calculateFoodDaysLeft(remainingFood, dailyFoodConsumption);
    cout << "Оставшейся пищи хватит на " << foodDaysLeft << " дней" << endl;

    return 0;
}

```

Вариант 6

Задание 1

Жители планеты Кор для того, чтобы избежать сдвига привычных времён года, ввели понятие високосного года (подобно тому, как это сделано в юлианском или григорианском земных календарях). В обычном Корском году ровно 180 Корских дней. Високосным является каждый 7-й год, а также каждый 9-й год. В такие годы добавляется один дополнительный день. Если же очередной 7-й год совпадает с очередным 9-м, то год считается обычным. Найдите точную продолжительность года на планете Кор (выраженную в Корских днях). Выберите из предложенных вариантов ответа, на сколько точная продолжительность года отличается от 180 дней.

Варианты ответа:

- А) 1/9
- Б) 2/9
- В) 2/7
- Г) 4/45

Ответ: Б) 2/9.

Задание 2

Космическое агентство разрабатывает автономного робота для исследования далёких планет. Робот должен уметь анализировать данные и выполнять повторяющиеся задачи в зависимости от полученных данных. Для этого разработана программа на языке C++, в которой используется несколько видов циклов и работа с массивами данных.

Входные данные:

- Робот получает данные о температуре на различных участках маршрута по поверхности планеты. Данные представлены в виде одномерного массива целых чисел (*temperatures[]*), где каждый элемент массива – это температура на конкретном участке.
- Робот фиксирует наличие препятствий на каждом участке, что представлено в виде другого массива (*obstacles[]*), где 1 означает наличие препятствия, а 0 – его отсутствие.

Роботу необходимо проанализировать данные и выполнить следующие задачи:

- 1) Робот должен двигаться по поверхности планеты, пока температура не опустится ниже -40°C или пока не встретит большое препятствие (обозначенное в массиве как 1). Определить, сколько участков робот пройдёт до остановки (переменная *count_before_stop*). Участок, на котором робот остановился, считать не нужно.
- 2) Робот должен проводить фотосъёмку каждого участка, где температура ниже 0°C, и определить, сколько снимков сделает робот до остановки (переменная *photo_count*). На участке, где остановился робот, можно провести фотосъёмку.
- 3) Робот должен подсчитать общее количество участков с нормальной температурой (от -40°C до +40°C) и без препятствий (переменная *count_normal_temp*).

Просуммируйте три числа, получившихся в результате выполнения указанных задач, и выберите из предложенных ниже вариантов ответа, чему равна эта сумма:

- A) 8
- Б) 9**
- B) 10
- Г) 7

Код программы:

```
#include <iostream>
```

```
using namespace std;
```

```
int main() {
```

```
    int temperatures[] = {-25, -40, 40, 0, -45, 30, -20};
```

```
    int obstacles[] = {0, 0, 0, 1, 0, 1, 0};
```

```
    int n = sizeof(temperatures) / sizeof(temperatures[0]);
```

```
    int i = 0;
```

```
    while (i < n && temperatures[i] >= -40 && obstacles[i] == 0) {
```

```
        i++;
```

```
    }
```

```
    int count_before_stop = i;
```

```
    int j = 0;
```

```
    int photo_count = 0;
```

```
    do {
```

```
        if (temperatures[j] < 0) {
```

```
            photo_count++;
```

```
        }
```

```
        j++;
```

```
    } while (j <= count_before_stop);
```

```
    int count_normal_temp = 0;
```

```
    for (int k = 0; k < n; k++) {
```

```
        if (temperatures[k] >= -40 && temperatures[k] <= 40 && obstacles[k] == 0) {
```

```
            count_normal_temp++;
```

```
        }
```

```

}

cout << "Количество участков до остановки: " << count_before_stop << endl;

cout << "Количество снимков: " << photo_count << endl;

cout << "Количество участков с нормальной температурой и без препятствий: " <<
count_normal_temp << endl;

return 0;

}

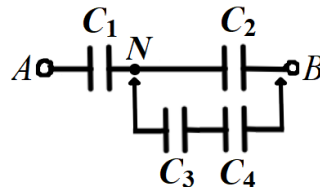
```

Ответ: Б) 9.

Задание 3

При проведении серии опытов на борту космической станции батарею конденсаторов C_1 и C_2 зарядили до разности потенциалов 13,3 В на клеммах $A-B$ (см. рисунок).

Затем источник отключили и к клеммам $N-B$ подключили батарею незаряженных



конденсаторов C_3 и C_4 . Ёмкости конденсаторов: $C_2 = 3,0$ мкФ, $C_3 = 4,0$ мкФ, $C_4 = 2,4$ мкФ. Площадь обкладок плоского конденсатора C_1 равна $1,2 \cdot 10^{-2}$ м², расстояние между ними $5,31 \cdot 10^{-7}$ м, диэлектрическая проницаемость вещества, заполняющего пространство между обкладками, $\epsilon = 42$. Вычислите максимальный заряд конденсатора C_4 .

Приставка 1мк = 10^{-6} ; $\epsilon_0 = 8,85 \cdot 10^{-12}$ Ф/м.

Варианты ответа:

- А) 4,5 мкКл
- Б) 9,8 мкКл**
- В) 29,4 мкКл
- Г) 8,4 мкКл
- Д) 12,0 мкКл

Ответ: Б). 9,8 мкКл

Задание 4

На поверхности неизвестной планеты по горизонтальной плоскости толкнули брусок с начальной скоростью 14 м/с. Коэффициент трения бруска о плоскость 0,26. Через некоторое расстояние S , скорость бруска уменьшилась на 6 м/с. Считая форму планеты шаром радиуса 170 км, первую космическую скорость для неё равной 600 м/с и учитывая

данные эксперимента, вычислите пройденное брусом расстояние S . Результат представьте в виде числа « $M \cdot 10^2$ м» и выберите ответ с мантиссой «М».

Гравитационная постоянная $G = 6,67 \cdot 10^{-11} \text{ Н} \cdot \text{м}^2/\text{кг}^2$.

Возможные варианты ответа:

- А) 7,8
- Б) 6,5
- В) 5,6
- Г) 4,3
- Д) 3,4
- Е) 2,1
- Ж) 1,2**

Ответ: Ж) 1,2.

Задание 5

Некоторая планета совершает полный оборот вокруг своей звезды за 544 часа, а сутки (для наблюдателя на этой планете) длятся 34 часа. За сколько часов эта планета совершает один оборот вокруг своей оси, если направление вращения вокруг оси и вокруг звезды совпадают? Выберите один вариант ответа из списка ниже с учётом округления.

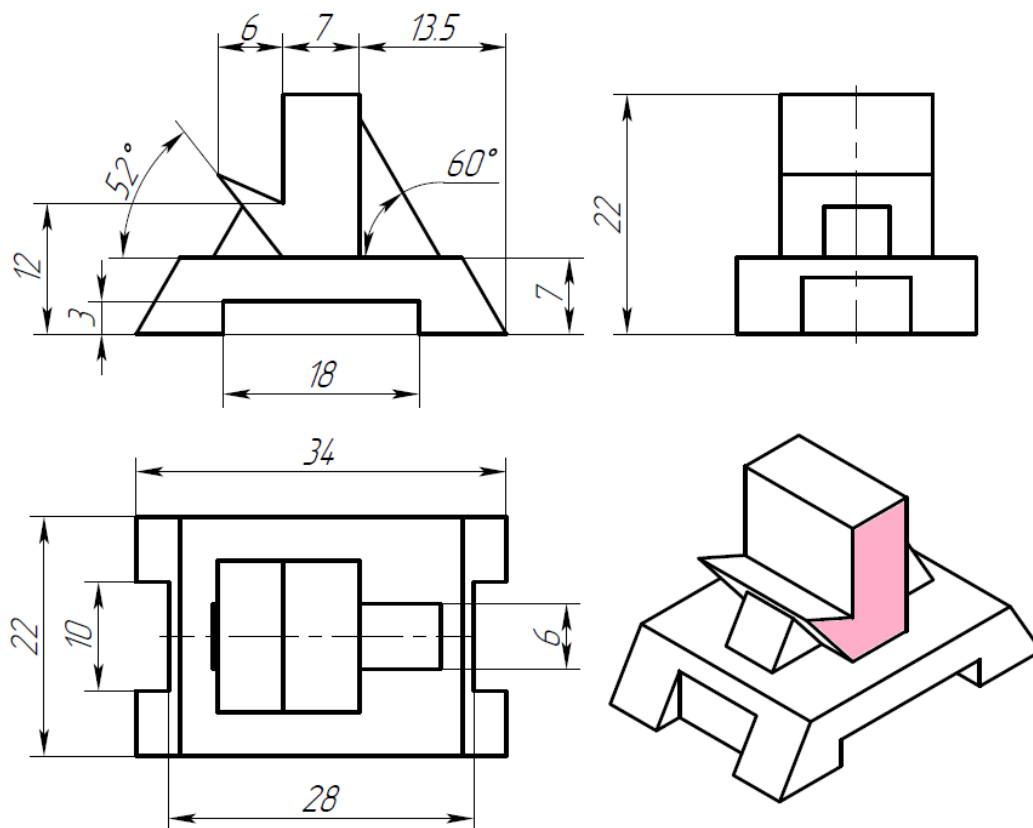
Варианты ответа:

- А) 36,27
- Б) 34,00
- В) 32,00**
- Г) 30,22

Ответ: В) 32,00.

Задание 6

По размерам, имеющимся на чертеже, определите площадь выделенной цветом грани.



Варианты ответа:

- А) 100
- Б) 110
- В) 120**
- Г) 130

Ответ: В) 120

Задание 7

Найти первую и вторую космические скорости для планеты Марс. Масса Марса составляет $6,43 \cdot 10^{23}$ кг, средний радиус – 3389,5 км. При расчётах принять гравитационную постоянную равной $6,67 \cdot 10^{-11}$ Н·м²/кг². Ответ выразите в км/с с одним знаком после запятой. При расчёте второй космической скорости необходимо воспользоваться полученным значением первой. В поля ответа необходимо ввести только числа.

Ответ: Первая космическая скорость: 3,6 км/с.

Вторая космическая скорость: 5,1 км/с.

Задание 8

Два тела движутся в одной плоскости. Если ввести на этой плоскости декартову систему координат, то траектории их движения совпадут с графиками функций $f(x) = 1 - x^2/8$ и $g(x) = \cos(2\pi x - \pi/4)$. Определите количество точек пересечения этих траекторий.

Ответ: 16.

Задание 9

Космический крейсер оказался под атакой космических пиратов и отправил сигнал SOS. Два отряда космодесантников, находящихся на базе, получили сигнал и направились на помощь.

От базы (узел А) до крейсера (узел Д) можно добраться через сеть червоточин, соединяющих некоторые узлы в пространстве (планеты или точки в пространстве).

Космическое пространство можно представить в виде графа с пятью вершинами, где вершины – это узлы (планеты или точки в пространстве), а рёбра – это червоточины с определённой длиной. Ниже в матрице смежности дана информация о длине рёбер-червоточин.

	А	Б	В	Г	Д
А	0	4	8	6	0
Б	4	0	7	2	9
В	8	7	0	1	5
Г	6	2	1	0	0
Д	0	9	5	0	0

Отряд Альфа обязан чередовать движение по рёбрам с чётной и нечётной длиной, при этом всегда начиная с четной.

Отряд Бета может перемещаться только по рёбрам с чётной длиной.

1) Какой из отрядов доберётся до крейсера быстрее? В случае равенства длин маршрутов выберите тот отряд, который прошёл через меньшее число узлов. В поле ответа введите название отряда одним словом с большой буквы без дополнительных символов (Альфа или Бета). **Ответ:** Альфа.

2) Возможно ли, что один из отрядов не сможет добраться до крейсера, учитывая

ограничения на рёбра? Напишите в поле ответа знак «+», если это возможно, или знак «-» – в противном случае (без пробелов, кавычек или иных посторонних символов). **Ответ:** +.

3) Введите в поле ответа длину кратчайшего маршрута (одним числом без пробелов и дополнительных знаков). **Ответ:** 13.

Задание 10

Космическое агентство разработало программу на языке C++ для управления уровнем углекислого газа на борту исследовательского космического корабля. Программа использует три функции, работающие с локальными и глобальными переменными, а также реализующими рекурсию.

Функция `calculateCO2Generation` принимает количество членов экипажа и количество дней полёта, возвращает количество углекислого газа, который экипаж произведет.

Функция `checkCO2Level` принимает текущий уровень углекислого газа и максимальный допустимый уровень. Если уровень CO2 превышает 75% от максимального, выводится предупреждение.

Рекурсивная функция `calculateDaysUntilMaxCO2` принимает начальный уровень углекислого газа и дневное его производство, возвращая количество дней до достижения максимального уровня.

Входные данные:

- Начальный уровень CO2 (переменная `CO2Level`): 50 единиц.
- Максимальный допустимый уровень CO2 (переменная `maxCO2Level`): 400 единиц.
- Суточное производство CO2 на одного члена экипажа (переменная `dailyCO2Production`): 5 единиц.
- Количество членов экипажа (переменная `crewMembers`): 4 человека.
- Количество дней полёта (переменная `flightDays`): 8 дней.

Вопросы:

1. Сколько углекислого газа произведет экипаж за 8 дней полёта? **Ответ:** 160.
2. Какой будет уровень CO2 после 8 дней полёта? **Ответ:** 210.
3. Сколько дней потребуется, чтобы достичь максимального уровня CO2? **Ответ:** 10.

Код программы:

```
#include <iostream>
using namespace std;
```

```
int CO2Level;
```

```
int calculateCO2Generation(int crewMembers, int days) {
```

```

    int CO2Produced = crewMembers * days * 5;
    return CO2Produced;
}

void checkCO2Level(int CO2, int maxCO2Level) {
    if (CO2 > 0.75 * maxCO2Level) {
        cout << "Внимание: Уровень CO2 превышает 75% от максимального!" << endl;
    }
}

int calculateDaysUntilMaxCO2(int CO2, int dailyCO2Production) {
    if (CO2 >= 400) {
        return 0;
    }
    return 1 + calculateDaysUntilMaxCO2(CO2 + dailyCO2Production * 4,
    dailyCO2Production);
}

int main() {
    CO2Level = 50;
    int crewMembers = 4;
    int flightDays = 8;
    int dailyCO2Production = 5;
    int maxCO2Level = 400;

    int CO2Produced = calculateCO2Generation(crewMembers, flightDays);
    cout << "Экипаж произведет " << CO2Produced << " единиц CO2 за " << flightDays
    << " дней" << endl;

    int remainingCO2 = CO2Level + CO2Produced;
    checkCO2Level(remainingCO2, maxCO2Level);
    cout << "Уровень CO2 после " << flightDays << " дней: " << remainingCO2 << "
    единиц" << endl;

    int daysUntilMaxCO2 = calculateDaysUntilMaxCO2(remainingCO2, dailyCO2Production);
    cout << "Максимального уровня CO2 достигнем через " << daysUntilMaxCO2 << "
    дней" << endl;

    return 0;
}

```

Вариант 7

Задание 1

Жители планеты Кор для того, чтобы избежать сдвига привычных времён года, ввели понятие високосного года (подобно тому, как это сделано в юлианском или григорианском земных календарях). В обычном Корском году ровно 400 Корских дней. Високосным является каждый 5-й год, а также каждый 8-й год. В такие годы добавляется один дополнительный день. Если же очередной 5-й год совпадает с очередным 8-м, то год считается обычным. Найдите точную продолжительность года на планете Кор (выраженную в Корских днях). Выберите из предложенных вариантов ответа, на сколько точная продолжительность года отличается от 400 дней.

Варианты ответа:

- А) 1/5
- Б) 1/2
- В) 11/40**
- Г) 7/200

Ответ: В) 11/40.

Задание 2

Космическое агентство разрабатывает автономного робота для исследования далёких планет. Робот должен уметь анализировать данные и выполнять повторяющиеся задачи в зависимости от полученных данных. Для этого разработана программа на языке C++, в которой используется несколько видов циклов и работа с массивами данных.

Входные данные:

- Робот получает данные о температуре на различных участках маршрута по поверхности планеты. Данные представлены в виде одномерного массива целых чисел (*temperatures[]*), где каждый элемент массива – это температура на конкретном участке.
- Робот фиксирует наличие препятствий на каждом участке, что представлено в виде другого массива (*obstacles[]*), где 1 означает наличие препятствия, а 0 – его отсутствие.

Роботу необходимо проанализировать данные и выполнить следующие задачи:

- 1) Робот должен двигаться по поверхности планеты, пока температура не опустится ниже -40°C или пока не встретит большое препятствие (обозначенное в массиве как 1). Определить, сколько участков робот пройдёт до остановки (переменная *count_before_stop*). Участок, на котором робот остановился, считать не нужно.
- 2) Робот должен проводить фотосъёмку каждого участка, где температура не выше 0°C, и определить, сколько снимков сделает робот до остановки (переменная *photo_count*). На участке, где остановился робот, можно провести фотосъёмку.
- 3) Робот должен подсчитать общее количество участков с нормальной температурой (от -40°C до +40°C) и без препятствий (переменная *count_normal_temp*).

Просуммируйте три числа, получившихся в результате выполнения указанных задач, и выберите из предложенных ниже вариантов ответа, чему равна эта сумма:

- A) 8
- Б) 9
- В) 10**
- Г) 7

Код программы:

```
#include <iostream>
```

```
using namespace std;
```

```
int main() {
```

```
    int temperatures[] = {-25, -40, 40, 0, -45, 30, -20};
```

```
    int obstacles[] = {0, 0, 0, 1, 0, 1, 0};
```

```
    int n = sizeof(temperatures) / sizeof(temperatures[0]);
```

```
    int i = 0;
```

```
    while (i < n && temperatures[i] >= -40 && obstacles[i] == 0) {
```

```
        i++;
```

```
    }
```

```
    int count_before_stop = i;
```

```
    int j = 0;
```

```
    int photo_count = 0;
```

```
    do {
```

```
        if (temperatures[j] <= 0) {
```

```
            photo_count++;
```

```
        }
```

```
        j++;
```

```
    } while (j <= count_before_stop);
```

```
    int count_normal_temp = 0;
```

```
    for (int k = 0; k < n; k++) {
```

```
        if (temperatures[k] >= -40 && temperatures[k] <= 40 && obstacles[k] == 0) {
```

```
            count_normal_temp++;
```

```
        }
```

```

}

cout << "Количество участков до остановки: " << count_before_stop << endl;

cout << "Количество снимков: " << photo_count << endl;

cout << "Количество участков с нормальной температурой и без препятствий: " <<
count_normal_temp << endl;

return 0;

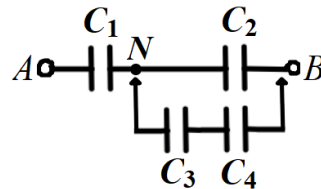
}

```

Ответ: В) 10.

Задание 3

При проведении серии опытов на борту космической станции к клеммам **A-B** батареи конденсаторов C_1 и C_2 подключили источник Э.Д.С. (см. рисунок).



Когда разность потенциалов на обкладках конденсатора C_1 достигла 6,75 В, источник отключили и к клеммам **N-B** подключили батарею незаряженных конденсаторов C_3 и C_4 . Ёмкости конденсаторов: $C_1 = 7$ мкФ, $C_3 = 16$ мкФ, $C_4 = 16$ мкФ. Площадь обкладок плоского конденсатора C_2 равна $4,2 \cdot 10^{-3}$ м², расстояние между ними $1,77 \cdot 10^{-7}$ м, диэлектрическая проницаемость вещества, заполняющего пространство между обкладками, $\epsilon = 20$. Рассчитайте максимальную суммарную энергию конденсаторов C_3 и C_4 .

Приставка 1мк = 10^{-6} ; $\epsilon_0 = 8,85 \cdot 10^{-12}$ Ф/м.

Варианты ответа:

- А) 4,2 мкДж
- Б) 47,25 мкДж
- В) 12,2 мкДж
- Г) 31 мкДж
- Д) **60 мкДж**

Ответ: Д) 60 мкДж

Задание 4

На поверхности неизвестной планеты вверх по наклонной плоскости под углом 30° к горизонту толкнули брусок с начальной скоростью 12 м/с. Коэффициент трения бруска о

плоскость 0,24. Через 22 м пути скорость бруска уменьшилась на Δv , но направление его движения не изменилось. Считая форму планеты шаром радиуса $2,22 \cdot 10^6$ м, первую космическую скорость для этой планеты равной 3,02 км/с и учитывая данные эксперимента, определите величину уменьшения скорости бруска $|\Delta v|$. Результат представьте в м/с.

Гравитационная постоянная $G = 6,67 \cdot 10^{-11} \text{ Н} \cdot \text{м}^2/\text{кг}^2$.

Возможные варианты ответа:

- А) 3 м/с
- Б) 4 м/с
- В) 5 м/с
- Г) 6 м/с
- Д) 7 м/с
- Е) 8 м/с**
- Ж) 9 м/с

Ответ: Е) 8 м/с

Задание 5

Некоторая планета совершает полный оборот вокруг своей звезды за 700 часов, а сутки (для наблюдателя на этой планете) длятся 35 часов. За сколько часов эта планета совершает один оборот вокруг своей оси, если направление вращения вокруг оси и вокруг звезды совпадают? Выберите один вариант ответа из списка ниже с учётом округления.

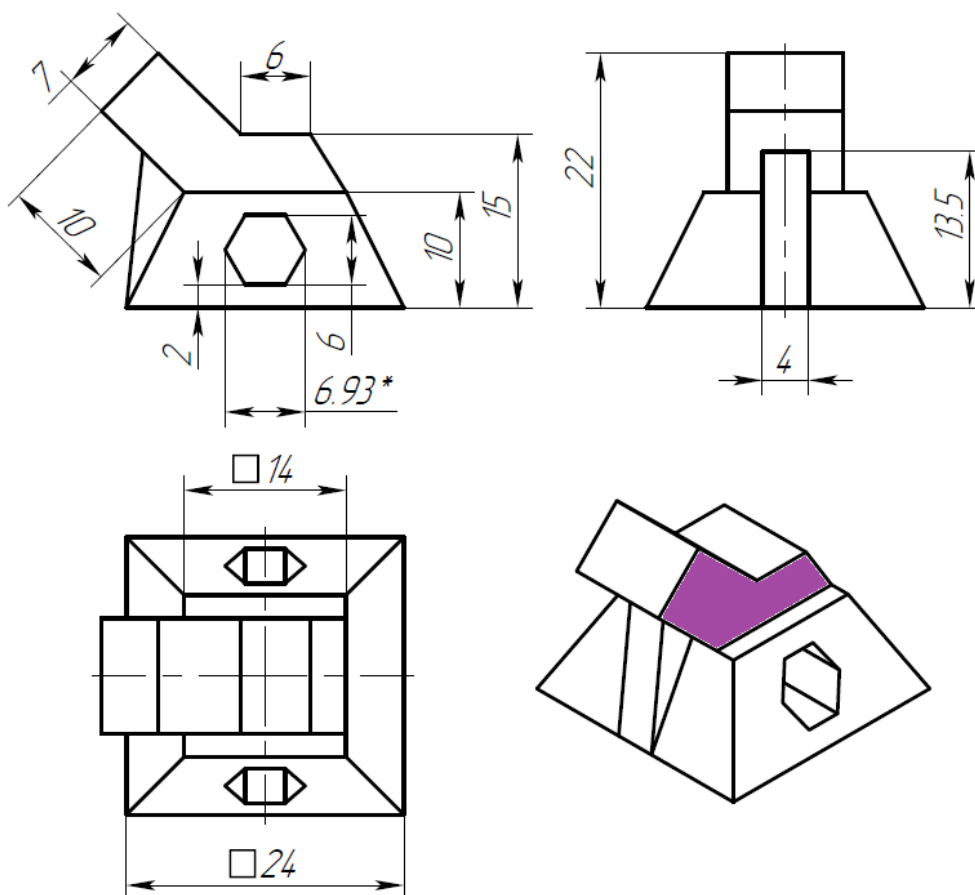
Варианты ответа:

- А) 35,00
- Б) 33,33**
- В) 36,84
- Г) 34,15

Ответ: Б) 33,33.

Задание 6

По размерам, имеющимся на чертеже, определите площадь выделенной цветом грани.



Варианты ответа:

- А) 100
- Б) 110
- В) 120**
- Г) 130

Ответ: В) 120

Задание 7

Найти первую и вторую космические скорости для планеты Венера. Масса Венеры составляет $4,88 \cdot 10^{24}$ кг, средний радиус – 6051,8 км. При расчётах принять гравитационную постоянную равной $6,67 \cdot 10^{-11}$ Н·м²/кг². Ответ выразите в км/с с одним знаком после запятой. При расчёте второй космической скорости необходимо воспользоваться полученным значением первой. В поля ответа необходимо ввести только числа.

Ответ: Первая космическая скорость: 7,3 км/с.

Вторая космическая скорость: 10,3 км/с.

Задание 8

Два тела движутся в одной плоскости. Если ввести на этой плоскости декартову систему координат, то траектории их движения совпадут с графиками функций $f(x) = 1 - \frac{(x-1)^2}{9}$ и $g(x) = \cos(\pi x)$. Определите количество точек пересечения этих траекторий.

Ответ: 10.

Задание 9

Космический крейсер оказался под атакой космических пиратов и отправил сигнал SOS. Два отряда космодесантников, находящихся на базе, получили сигнал и направились на помощь.

От базы (узел А) до крейсера (узел Д) можно добраться через сеть червоточин, соединяющих некоторые узлы в пространстве (планеты или точки в пространстве).

Космическое пространство можно представить в виде графа с пятью вершинами, где вершины – это узлы (планеты или точки в пространстве), а рёбра – это червоточины с определённой длиной. Ниже в матрице смежности дана информация о длине рёбер-червоточин.

	А	Б	В	Г	Д
А	0	3	8	6	0
Б	3	0	5	2	9
В	8	5	0	1	7
Г	6	2	1	0	6
Д	0	9	7	6	0

Отряд Альфа обязан чередовать движение по рёбрам с чётной и нечётной длиной, при этом чётность длины первого ребра не важна.

Отряд Бета может перемещаться только по рёбрам с чётной длиной.

1) Какой из отрядов доберётся до крейсера быстрее? В случае равенства длин маршрутов выберите тот отряд, который прошёл через меньшее число узлов. В поле ответа введите название отряда одним словом с большой буквы без дополнительных символов (Альфа или Бета). **Ответ:** Бета.

2) Возможно ли, что один из отрядов не сможет добраться до крейсера, учитывая

ограничения на рёбра? Напишите в поле ответа знак «+», если это возможно, или знак «-» – в противном случае (без пробелов, кавычек или иных посторонних символов). **Ответ:** -.

3) Введите в поле ответа длину кратчайшего маршрута (одним числом без пробелов и дополнительных знаков). **Ответ:** 12.

Задание 10

Космическое агентство разработало программу на языке C++ для управления временем работы системы жизнеобеспечения на борту исследовательского космического корабля. Программа использует три функции, работающие с локальными и глобальными переменными, а также реализующими рекурсию.

Функция `calculatePowerConsumption` принимает мощность потребления системы за сутки и количество дней полёта, возвращает общее количество энергии, которую система израсходует.

Функция `checkPowerLevel` принимает текущий уровень энергии и максимальный уровень. Если уровень энергии падает ниже 25% от максимального значения, выводится предупреждение.

Рекурсивная функция `calculateDaysUntilPowerDepletion` принимает текущий уровень энергии и суточное потребление энергии, возвращая количество дней, на которые хватит оставшегося запаса.

Входные данные:

- Начальный уровень энергии (переменная `powerLevel`): 10000 единиц.
- Максимальный уровень энергии (переменная `maxPowerLevel`): 10000 единиц.
- Суточное потребление системы (переменная `dailyPowerConsumption`): 600 единиц.
- Количество дней полёта (переменная `flightDays`): 12 дней.

Вопросы:

1. Сколько энергии израсходует система за 12 дней полёта? **Ответ:** 7200.
2. Какой будет уровень энергии после 12 дней полёта? **Ответ:** 2800.
3. На сколько дней хватит оставшегося запаса энергии? **Ответ:** 4.

Код программы:

```
#include <iostream>
using namespace std;

int powerLevel;

int calculatePowerConsumption(int dailyConsumption, int days) {
    int powerConsumed = dailyConsumption * days;
    return powerConsumed;
}

void checkPowerLevel(int power, int maxPower) {
```

```

    if (power < 0.25 * maxPower) {
        cout << "Внимание: Уровень энергии ниже 25%!" << endl;
    }
}

int calculateDaysUntilPowerDepletion(int power, int dailyConsumption) {
    if (power < dailyConsumption) {
        return 0;
    }
    return 1 + calculateDaysUntilPowerDepletion(power - dailyConsumption,
dailyConsumption);
}

int main() {
    powerLevel = 10000;
    int dailyPowerConsumption = 600;
    int flightDays = 12;
    int maxPowerLevel = 10000;

    int powerConsumed = calculatePowerConsumption(dailyPowerConsumption, flightDays);
    cout << "Система израсходует " << powerConsumed << " единиц энергии за " <<
flightDays << " дней" << endl;

    int remainingPower = powerLevel - powerConsumed;
    checkPowerLevel(remainingPower, maxPowerLevel);
    cout << "Уровень энергии после " << flightDays << " дней: " << remainingPower << "
единиц" << endl;
    int daysUntilPowerDepletion = calculateDaysUntilPowerDepletion(remainingPower,
dailyPowerConsumption);
    cout << "Оставшейся энергии хватит на " << daysUntilPowerDepletion << " дней" <<
endl;

    return 0;
}

```

Вариант 8

Задание 1

Жители планеты Кор для того, чтобы избежать сдвига привычных времён года, ввели понятие високосного года (подобно тому, как это сделано в юлианском или григорианском земных календарях). В обычном Корском году ровно 355 Корских дней. Високосным является каждый 4-й год, а также каждый 7-й год. В такие годы добавляется один дополнительный день. Если же очередной 4-й год совпадает с очередным 7-м, то год считается обычным. Найдите точную продолжительность года на планете Кор (выраженную в Корских днях). Выберите из предложенных вариантов ответа, на сколько точная продолжительность года отличается от 355 дней.

Варианты ответа:

- А) 1/6
- Б) 5/14
- В) 9/28**
- Г) 28/355

Ответ: В) 9/28.

Задание 2

Космическое агентство разрабатывает автономного робота для исследования далёких планет. Робот должен уметь анализировать данные и выполнять повторяющиеся задачи в зависимости от полученных данных. Для этого разработана программа на языке C++, в которой используется несколько видов циклов и работа с массивами данных.

Входные данные:

- Робот получает данные о температуре на различных участках маршрута по поверхности планеты. Данные представлены в виде одномерного массива целых чисел (*temperatures[]*), где каждый элемент массива – это температура на конкретном участке.
- Робот фиксирует наличие препятствий на каждом участке, что представлено в виде другого массива (*obstacles[]*), где 1 означает наличие препятствия, а 0 – его отсутствие.

Роботу необходимо проанализировать данные и выполнить следующие задачи:

- 1) Робот должен двигаться по поверхности планеты, пока температура не опустится ниже -40°C или пока не встретит большое препятствие (обозначенное в массиве как 1). Определить, сколько участков робот пройдёт до остановки (переменная *count_before_stop*). Участок, на котором робот остановился, считать не нужно.
- 2) Робот должен проводить фотосъёмку каждого участка, где температура не ниже 0°C, и определить, сколько снимков сделает робот до остановки (переменная *photo_count*). На участке, где остановился робот, можно провести фотосъёмку.
- 3) Робот должен подсчитать общее количество участков с нормальной температурой (от -40°C до +40°C) и без препятствий (переменная *count_normal_temp*).

Просуммируйте три числа, получившихся в результате выполнения указанных задач, и выберите из предложенных ниже вариантов ответа, чему равна эта сумма:

- A) 8
- Б) 9
- В) 7
- Г) 5

Код программы:

```
#include <iostream>
```

```
using namespace std;
```

```
int main() {
```

```
    int temperatures[] = {-25, -40, -40, 0, -45, 30, -20};
```

```
    int obstacles[] = {0, 0, 1, 1, 0, 1, 0};
```

```
    int n = sizeof(temperatures) / sizeof(temperatures[0]);
```

```
    int i = 0;
```

```
    while (i < n && temperatures[i] >= -40 && obstacles[i] == 0) {
```

```
        i++;
```

```
    }
```

```
    int count_before_stop = i;
```

```
    int j = 0;
```

```
    int photo_count = 0;
```

```
    do {
```

```
        if (temperatures[j] >= 0) {
```

```
            photo_count++;
```

```
        }
```

```
        j++;
```

```
    } while (j <= count_before_stop);
```

```
    int count_normal_temp = 0;
```

```
    for (int k = 0; k < n; k++) {
```

```
        if (temperatures[k] >= -40 && temperatures[k] <= 40 && obstacles[k] == 0) {
```

```
            count_normal_temp++;
```

```
        }
```

```

}

cout << "Количество участков до остановки: " << count_before_stop << endl;

cout << "Количество снимков: " << photo_count << endl;

cout << "Количество участков с нормальной температурой и без препятствий: " <<
count_normal_temp << endl;

return 0;

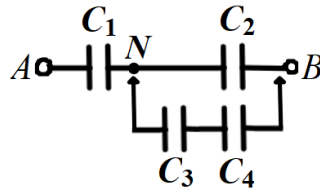
}

```

Ответ: Г) 5.

Задание 3

При проведении серии опытов на борту космической станции к клеммам *A-B* батареи конденсаторов C_1 и C_2 подключили источник Э.Д.С. (см. рисунок).



Когда конденсатор C_1 стал обладать энергией 38,5 мкДж, источник отключили и к клеммам *N-B* подключили батарею незаряженных конденсаторов C_3 и C_4 . Ёмкости конденсаторов: $C_2 = 1,8$ мкФ, $C_3 = 7,8$ мкФ, $C_4 = 2,7$ мкФ. Площадь обкладок плоского конденсатора C_1 равна $6 \cdot 10^{-3}$ м², расстояние между ними $5,31 \cdot 10^{-7}$ м, диэлектрическая проницаемость вещества, заполняющего пространство между обкладками, $\epsilon = 42$. Рассчитайте заряд конденсатора C_2 после максимальной зарядки конденсаторов C_3 и C_4 .

Приставка 1мк = 10^{-6} ; $\epsilon_0 = 8,85 \cdot 10^{-12}$ Ф/м.

Варианты ответа:

- А) 1,4 мкКл
- Б) 14,3 мкКл
- В) 8,5 мкКл**
- Г) 7,0 мкКл
- Д) 2,2 мкКл

Ответ: В). 8,5 мкКл

Задание 4

На поверхности неизвестной планеты по горизонтальной плоскости толкнули брусок с начальной скоростью v_0 . Коэффициент трения бруска о плоскость 0,26. Через 57 м пути,

скорость бруска уменьшилась на 8 м/с. Считая форму планеты шаром радиуса $7 \cdot 10^6$ м, первую космическую скорость для неё равной 5,5 км/с и учитывая данные эксперимента, вычислите начальную скорость бруска v_0 . Результат представьте в м/с.

Гравитационная постоянная $G = 6,67 \cdot 10^{-11} \text{ Н} \cdot \text{м}^2/\text{кг}^2$.

Возможные варианты ответа:

- А) 9 м/с
- Б) 10 м/с
- В) 11 м/с
- Г) 12 м/с**
- Д) 13 м/с
- Е) 14 м/с
- Ж) 15 м/с

Ответ: Г) 12 м/с

Задание 5

Некоторая планета совершает полный оборот вокруг своей звезды за 840 часов, а сутки (для наблюдателя на этой планете) длятся 42 часа. За сколько часов эта планета совершает один оборот вокруг своей оси, если направление вращения вокруг оси и вокруг звезды совпадают? Выберите один вариант ответа из списка ниже с учётом округления.

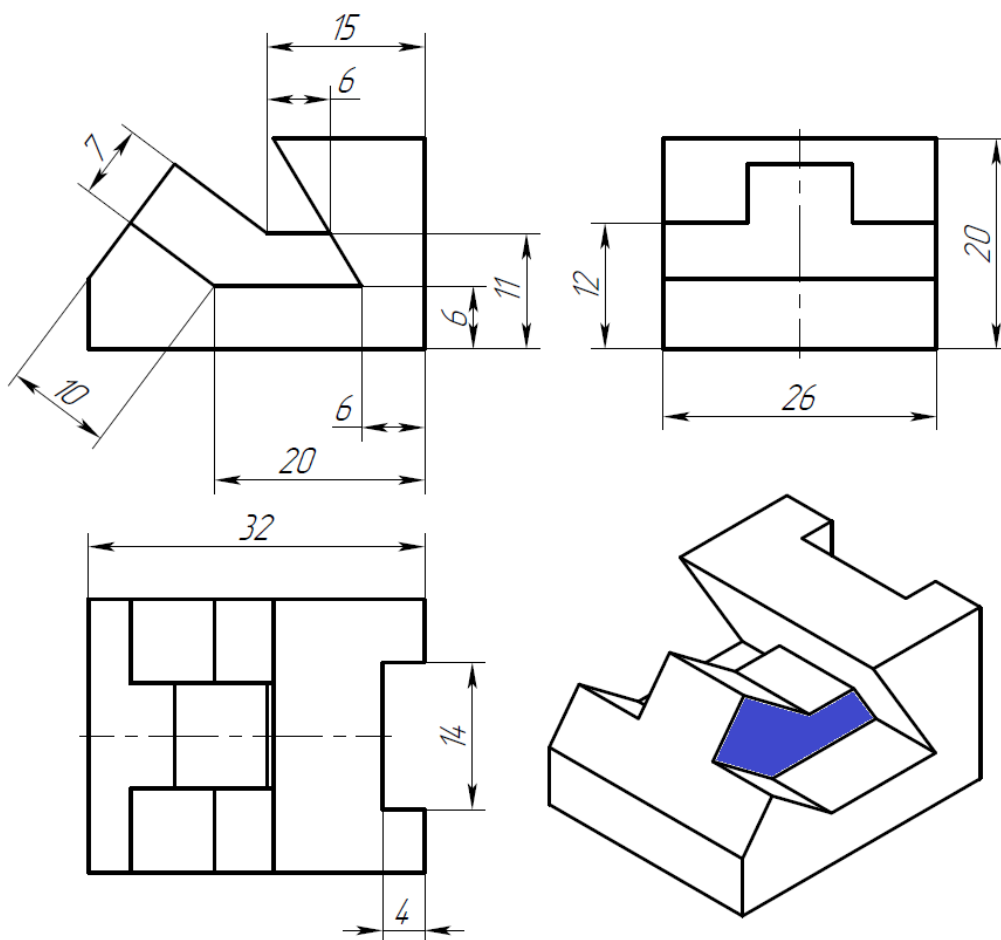
Варианты ответа:

- А) 42,00
- Б) 38,18
- В) 44,21
- Г) 40,00**

Ответ: Г) 40,00.

Задание 6

По размерам, имеющимся на чертеже, определите площадь выделенной цветом грани.



Варианты ответа:

- А) 100
- Б) 110
- В) 120**
- Г) 130

Ответ: В) 120

Задание 7

Найти первую и вторую космические скорости для планеты Юпитер. Масса Юпитера составляет $1,9 \cdot 10^{27}$ кг, средний радиус – 7130 км. При расчётах принять гравитационную постоянную равной $6,67 \cdot 10^{-11}$ Н·м²/кг². Ответ выразите в км/с с одним знаком после запятой. При расчёте второй космической скорости необходимо воспользоваться полученным значением первой. В поля ответа необходимо ввести только числа.

Ответ: Первая космическая скорость: 133,3 км/с.

Вторая космическая скорость: 188,5 км/с.

Задание 8

Два тела движутся в одной плоскости. Если ввести на этой плоскости декартову систему координат, то траектории их движения совпадут с графиками функций $f(x) = 1 - \frac{(x-0.5)^2}{13}$ и $g(x) = \cos(\pi x)$. Определите количество точек пересечения этих траекторий.

Ответ: 10.

Задание 9

Космический крейсер оказался под атакой космических пиратов и отправил сигнал SOS. Два отряда космодесантников, находящихся на базе, получили сигнал и направились на помощь.

От базы (узел А) до крейсера (узел Д) можно добраться через сеть червоточин, соединяющих некоторые узлы в пространстве (планеты или точки в пространстве).

Космическое пространство можно представить в виде графа с пятью вершинами, где вершины – это узлы (планеты или точки в пространстве), а рёбра – это червоточины с определённой длиной. Ниже в матрице смежности дана информация о длине рёбер-червоточин.

	А	Б	В	Г	Д
А	0	6	4	8	0
Б	6	0	7	2	9
В	4	7	0	1	5
Г	8	2	1	0	6
Д	0	9	5	6	0

Отряд Альфа обязан чередовать движение по рёбрам с чётной и нечётной длиной, при этом чётность длины первого ребра не важна.

Отряд Бета может перемещаться только по рёбрам с чётной длиной.

1) Какой из отрядов доберётся до крейсера быстрее? В случае равенства длин маршрутов выберите тот отряд, который прошёл через меньшее число узлов. В поле ответа введите название отряда одним словом с большой буквы без дополнительных символов (Альфа или Бета). **Ответ:** Альфа.

2) Возможно ли, что один из отрядов не сможет добраться до крейсера, учитывая

ограничения на рёбра? Напишите в поле ответа знак «+», если это возможно, или знак «-» – в противном случае (без пробелов, кавычек или иных посторонних символов). **Ответ:** -.

3) Введите в поле ответа длину кратчайшего маршрута (одним числом без пробелов и дополнительных знаков). **Ответ:** 9.

Задание 10

Космическое агентство разработало программу на языке C++ для управления запасом жидкости на борту исследовательского космического корабля. Программа использует три функции, работающие с локальными и глобальными переменными, а также реализующие рекурсию.

Функция `calculateLiquidConsumption` принимает количество членов экипажа и количество дней полёта, возвращает общее количество жидкости, которое израсходует экипаж.

Функция `checkLiquidLevel` принимает текущий уровень жидкости и минимально допустимый уровень. Если уровень жидкости ниже 30% от начального запаса, выводится предупреждение.

Рекурсивная функция `calculateLiquidDaysLeft` принимает текущий запас жидкости, суточное потребление жидкости на одного члена экипажа и количество членов экипажа, возвращает количество дней, на которые хватит оставшегося запаса жидкости.

Входные данные:

- Начальный запас жидкости (переменная `liquidLevel`): 800 литров.
- Минимально допустимый уровень жидкости (переменная `minLiquidLevel`): 30% от начального запаса.
- Суточное потребление жидкости на одного члена экипажа (переменная `dailyLiquidConsumption`): 2 литра.
- Количество членов экипажа (переменная `crewMembers`): 6 человек.
- Количество дней полёта (переменная `flightDays`): 7 дней.

Вопросы:

1. Сколько жидкости израсходует экипаж за 7 дней полёта? **Ответ:** 84.
2. Какой будет уровень жидкости после 7 дней полёта? **Ответ:** 716.
3. На сколько дней хватит оставшегося запаса жидкости? **Ответ:** 59.

Код программы:

```
#include <iostream>
using namespace std;
```

```
int liquidLevel;
```

```
int calculateLiquidConsumption(int crewMembers, int days) {
```

```

    int liquidConsumed = crewMembers * days * 2;
    return liquidConsumed;
}

void checkLiquidLevel(int liquid, int minLiquidLevel) {
    if (liquid < minLiquidLevel) {
        cout << "Внимание: Уровень жидкости ниже допустимого!" << endl;
    }
}

int calculateLiquidDaysLeft(int liquid, int dailyLiquidConsumption, int crewMembers) {
    if (liquid < dailyLiquidConsumption * crewMembers) {
        return 0;
    }
    return 1 + calculateLiquidDaysLeft(liquid - dailyLiquidConsumption * crewMembers,
    dailyLiquidConsumption, crewMembers);
}

int main() {
    liquidLevel = 800;
    int crewMembers = 6;
    int dailyLiquidConsumption = 2;
    int flightDays = 7;
    int minLiquidLevel = 0.3 * liquidLevel;

    int liquidConsumed = calculateLiquidConsumption(crewMembers, flightDays);
    cout << "Экипаж израсходует " << liquidConsumed << " литров жидкости за " <<
    flightDays << " дней" << endl;

    int remainingLiquid = liquidLevel - liquidConsumed;
    checkLiquidLevel(remainingLiquid, minLiquidLevel);
    cout << "Уровень жидкости после " << flightDays << " дней: " << remainingLiquid << "
    литров" << endl;

    int liquidDaysLeft = calculateLiquidDaysLeft(remainingLiquid, dailyLiquidConsumption,
    crewMembers);
    cout << "Оставшегося запаса жидкости хватит на " << liquidDaysLeft << " дней" <<
    endl;

    return 0;
}

```

Вариант 9

Задание 1

Жители планеты Марь для того, чтобы избежать сдвига привычных времён года, ввели понятие антивисокосного года, который на один день короче обычного. В обычном Марьском году ровно 150 Марьских дней. Антивисокосным является каждый 5-й год, а также каждый 11-й год. Если же очередной 5-й год совпадает с очередным 11-м, то убирается только один день. Найдите точную продолжительность года на планете Марь (выраженную в Марьских днях). Выберите из предложенных вариантов ответа, на сколько точная продолжительность года отличается от 150 дней.

Варианты ответа:

- А) 8/55
- Б) 1/10
- В) 8/75
- Г) 3/11

Ответ: Г) 3/11.

Задание 2

Космическое агентство разрабатывает автономного робота для исследования далёких планет. Робот должен уметь анализировать данные и выполнять повторяющиеся задачи в зависимости от полученных данных. Для этого разработана программа на языке C++, в которой используется несколько видов циклов и работа с массивами данных.

Входные данные:

- Робот получает данные о температуре на различных участках маршрута по поверхности планеты. Данные представлены в виде одномерного массива целых чисел (*temperatures[]*), где каждый элемент массива – это температура на конкретном участке.
- Робот фиксирует наличие препятствий на каждом участке, что представлено в виде другого массива (*obstacles[]*), где 1 означает наличие препятствия, а 0 – его отсутствие.

Роботу необходимо проанализировать данные и выполнить следующие задачи:

- 1) Робот должен двигаться по поверхности планеты, пока температура не поднимется выше 40°C или пока не встретит большое препятствие (обозначенное в массиве как 1). Определить, сколько участков робот пройдёт до остановки (переменная *count_before_stop*). Участок, на котором робот остановился, считать не нужно.
- 2) Робот должен проводить фотосъёмку каждого участка, где температура выше 0°C, и определить, сколько снимков сделает робот до остановки (переменная *photo_count*). На участке, где остановился робот, можно провести фотосъёмку.
- 3) Робот должен подсчитать общее количество участков с нормальной температурой (от -35°C до +35°C) и без препятствий (переменная *count_normal_temp*).

Просуммируйте три числа, получившихся в результате выполнения указанных задач, и выберите из предложенных ниже вариантов ответа, чему равна эта сумма:

- A) 9
- Б) 10
- В) 11
- Г) 12

Код программы:

```
#include <iostream>
```

```
using namespace std;
```

```
int main() {
```

```
    int temperatures[] = {-25, -40, 40, 0, -45, 30, -20};
```

```
    int obstacles[] = {0, 0, 0, 0, 0, 1, 1};
```

```
    int n = sizeof(temperatures) / sizeof(temperatures[0]);
```

```
    int i = 0;
```

```
    while (i < n && temperatures[i] <= 40 && obstacles[i] == 0) {
```

```
        i++;
```

```
    }
```

```
    int count_before_stop = i;
```

```
    int j = 0;
```

```
    int photo_count = 0;
```

```
    do {
```

```
        if (temperatures[j] > 0) {
```

```
            photo_count++;
```

```
        }
```

```
        j++;
```

```
    } while (j <= count_before_stop);
```

```
    int count_normal_temp = 0;
```

```
    for (int k = 0; k < n; k++) {
```

```
        if (temperatures[k] >= -35 && temperatures[k] <= 35 && obstacles[k] == 0) {
```

```
            count_normal_temp++;
```

```
        }
```

```

}

cout << "Количество участков до остановки: " << count_before_stop << endl;

cout << "Количество снимков: " << photo_count << endl;

cout << "Количество участков с нормальной температурой и без препятствий: " <<
count_normal_temp << endl;

return 0;

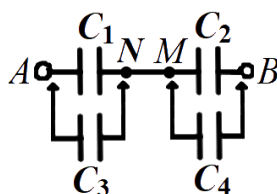
}

```

Ответ: А) 9.

Задание 3

При проведении серии опытов на борту космической станции к клеммам **A-B** батареи конденсаторов C_1 и C_2 подключили источник Э.Д.С. (см. рисунок).



Когда конденсатор C_2 стал обладать энергией 3,1 мкДж, источник отключили и к блоку C_{12} подключили незаряженные конденсаторы C_3 и C_4 . Схема их подключения показана на рисунке. Ёмкости конденсаторов $C_1 = 6,3$ мкФ, $C_3 = 24,0$ мкФ, $C_4 = 6,0$ мкФ. Площадь обкладок плоского конденсатора C_2 равна $0,261$ м², расстояние между ними $8,85 \cdot 10^{-7}$ м, диэлектрическая проницаемость вещества, заполняющего пространство между обкладками, $\varepsilon = 74$. Рассчитайте суммарную энергию конденсаторов C_1 и C_2 после максимальной зарядки конденсаторов C_3 и C_4 .

Приставка $1\text{мк} = 10^{-6}$; $\varepsilon_0 = 8,85 \cdot 10^{-12}$ Ф/м.

Варианты ответа:

- А) 1,5 мкДж
- Б) 7,0 мкДж**
- В) 2,1 мкДж
- Г) 0,6 мкДж
- Д) 9,2 мкДж

Ответ: Б) 7,0 мкДж

Задание 4

С поверхности неизвестной планеты вертикально вверх толкнули ядро с начальной скоростью 10 м/с. Через 3 секунды его скорость стала равна 0,8 м/с и направлена вниз. Учитывая результаты данного эксперимента, считая форму планеты шаром радиуса 100 км, вычислите скорость спутника, движущегося по круговой орбите на высоте 125 км над

её поверхностью. Результат представьте в виде числа « $M \cdot 10^2$ м/с» и выберите ответ с мантиссой «М».

Гравитационная постоянная $G = 6,67 \cdot 10^{-11} \text{ Н} \cdot \text{м}^2/\text{кг}^2$.

Возможные варианты ответа:

- А) 3
- Б) 4**
- В) 5
- Г) 6
- Д) 7
- Е) 8
- Ж) 9

Ответ: Б) 4.

Задание 5

Некоторая планета совершает полный оборот вокруг своей звезды за 400 часов, а сутки (для наблюдателя на этой планете) длятся 50 часов. За сколько часов эта планета совершает один оборот вокруг своей оси, если направление вращения вокруг оси противоположно направлению вращения вокруг звезды? Выберите один вариант ответа из списка ниже с учётом округления.

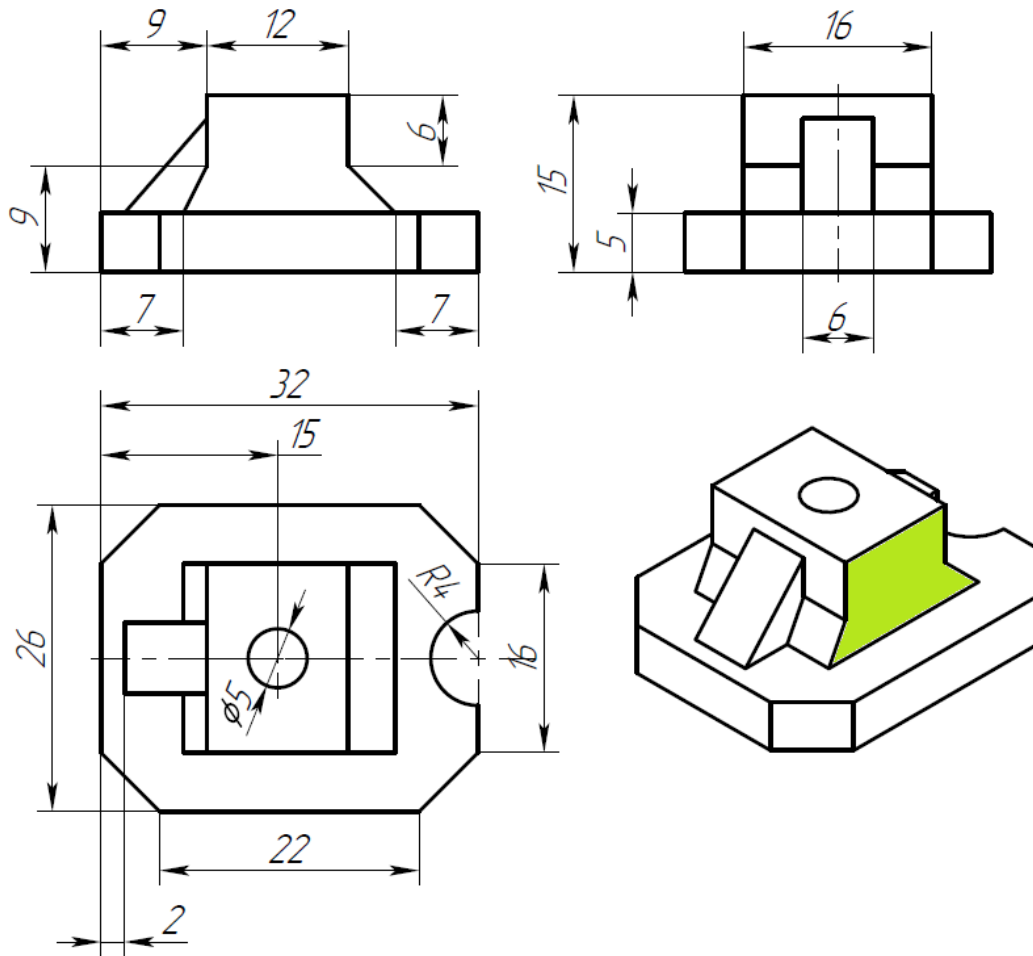
Варианты ответа:

- А) 44,44
- Б) 50,00
- В) 57,14**
- Г) 47,06

Ответ: В) 57,14.

Задание 6

По размерам, имеющимся на чертеже, определите площадь выделенной цветом грани.



Варианты ответа:

- А) 128
- Б) 132**
- В) 124
- Г) 136

Ответ: Б) 132

Задание 7

Найти первую и вторую космические скорости для планеты Уран. Диаметр Урана – 47 100 км, ускорение свободного падения на поверхности составляет $8,86 \text{ м/с}^2$. Ответ выразите в км/с с одним знаком после запятой. При расчёте второй космической скорости необходимо воспользоваться полученным значением первой. В поля ответа необходимо ввести только числа.

Ответ: Первая космическая скорость: 14,4 км/с.

Вторая космическая скорость: 20,4 км/с.

Задание 8

Два тела движутся в одной плоскости. Если ввести на этой плоскости декартову систему координат, то траектории их движения совпадут с графиками функций $f(x) = x^2/13 - 1$ и $g(x) = \sin(\pi x + \pi/3)$. Определите количество точек пересечения этих траекторий.

Ответ: 10.

Задание 9

Космический крейсер оказался под атакой космических пиратов и отправил сигнал SOS. Два отряда космодесантников, находящихся на базе, получили сигнал и направились на помощь.

От базы (узел А) до крейсера (узел Д) можно добраться через сеть червоточин, соединяющих некоторые узлы в пространстве (планеты или точки в пространстве).

Космическое пространство можно представить в виде графа с пятью вершинами, где вершины – это узлы (планеты или точки в пространстве), а рёбра – это червоточины с определённой длиной. Ниже в матрице смежности дана информация о длине рёбер-червоточин.

	А	Б	В	Г	Д
А	0	4	7	6	0
Б	4	0	2	6	8
В	7	2	0	1	9
Г	6	6	1	0	0
Д	0	8	9	0	0

Отряд Альфа обязан начать движение с ребра с чётной длиной и чередовать движение по рёбрам с нечётной и чётной длиной.

Отряд Бета может перемещаться только по рёбрам с нечётной длиной.

1) Какой из отрядов доберётся до крейсера быстрее? В случае равенства длин маршрутов выберите тот отряд, который прошёл через меньшее число узлов. В поле ответа введите название отряда одним словом с большой буквы без дополнительных символов (Альфа или Бета). **Ответ:** Бета.

2) Возможно ли, что один из отрядов не сможет добраться до крейсера, учитывая

ограничения на рёбра? Напишите в поле ответа знак «+», если это возможно, или знак «-» – в противном случае (без пробелов, кавычек или иных посторонних символов). **Ответ:** +.

3) Введите в поле ответа длину кратчайшего маршрута (одним числом без пробелов и дополнительных знаков). **Ответ:** 16.

Задание 10

Космическое агентство разработало программу на языке C++ для управления запасами пищи на борту исследовательского космического корабля. Программа использует три функции, работающие с локальными и глобальными переменными, а также реализующие рекурсию.

Функция `calculateFoodConsumption` принимает количество членов экипажа и количество дней полёта, возвращает общее количество пищи, которое израсходует экипаж.

Функция `checkFoodLevel` принимает текущий уровень пищи и минимально допустимый уровень. Если уровень пищи падает ниже 20% от начального запаса, выводится предупреждение.

Рекурсивная функция `calculateFoodDaysLeft` принимает текущий запас пищи, суточное потребление пищи на одного члена экипажа и количество членов экипажа, возвращает количество дней, на которые хватит оставшегося запаса пищи.

Входные данные:

- Начальный запас пищи (переменная `foodLevel`): 600 кг.
- Минимально допустимый уровень пищи (переменная `minFoodLevel`): 20% от начального запаса.
- Суточное потребление пищи на одного члена экипажа (переменная `dailyFoodConsumption`): 4 кг.
- Количество членов экипажа (переменная `crewMembers`): 5 человек.
- Количество дней полёта (переменная `flightDays`): 10 дней.

Вопросы:

1. Сколько пищи израсходует экипаж за 10 дней полёта? **Ответ:** 200.
2. Какой будет уровень пищи после 10 дней полёта? **Ответ:** 400.
3. На сколько дней хватит оставшегося запаса пищи? **Ответ:** 20.

Код программы:

```
#include <iostream>
using namespace std;
```

```
int foodLevel;
```

```
int calculateFoodConsumption(int crewMembers, int days) {
```

```

    int foodConsumed = crewMembers * days * 4;
    return foodConsumed;
}

void checkFoodLevel(int food, int minFoodLevel) {
    if (food < minFoodLevel) {
        cout << "Внимание: Уровень пищи ниже допустимого!" << endl;
    }
}

int calculateFoodDaysLeft(int food, int dailyFoodConsumption, int crewMembers) {
    if (food < dailyFoodConsumption * crewMembers) {
        return 0;
    }
    return 1 + calculateFoodDaysLeft(food - dailyFoodConsumption * crewMembers,
    dailyFoodConsumption, crewMembers);
}

int main() {
    foodLevel = 600;
    int crewMembers = 5;
    int dailyFoodConsumption = 4;
    int flightDays = 10;
    int minFoodLevel = 0.2 * foodLevel;

    int foodConsumed = calculateFoodConsumption(crewMembers, flightDays);
    cout << "Экипаж израсходует " << foodConsumed << " кг пищи за " << flightDays << "
    дней" << endl;

    int remainingFood = foodLevel - foodConsumed;
    checkFoodLevel(remainingFood, minFoodLevel);
    cout << "Уровень пищи после " << flightDays << " дней: " << remainingFood << " кг"
    << endl;

    int foodDaysLeft = calculateFoodDaysLeft(remainingFood, dailyFoodConsumption,
    crewMembers);
    cout << "Оставшегося запаса пищи хватит на " << foodDaysLeft << " дней" << endl;

    return 0;
}

```

Вариант 10

Задание 1

Жители планеты Марь для того, чтобы избежать сдвига привычных времён года, ввели понятие антивисокосного года, который на один день короче обычного. В обычном Марьском году ровно 210 Марьских дней. Антивисокосным является каждый 7-й год, а также каждый 9-й год. Если же очередной 7-й год совпадает с очередным 9-м, то убирается только один день. Найдите точную продолжительность года на планете Марь (выраженную в Марьских днях). Выберите из предложенных вариантов ответа, на сколько точная продолжительность года отличается от 210 дней.

Варианты ответа:

- А) 5/21
- Б) 1/4
- В) 16/63
- Г) 16/210

Ответ: А) 5/21

Задание 2

Космическое агентство разрабатывает автономного робота для исследования далёких планет. Робот должен уметь анализировать данные и выполнять повторяющиеся задачи в зависимости от полученных данных. Для этого разработана программа на языке C++, в которой используется несколько видов циклов и работа с массивами данных.

Входные данные:

- Робот получает данные о температуре на различных участках маршрута по поверхности планеты. Данные представлены в виде одномерного массива целых чисел (*temperatures[]*), где каждый элемент массива – это температура на конкретном участке.
- Робот фиксирует наличие препятствий на каждом участке, что представлено в виде другого массива (*obstacles[]*), где 1 означает наличие препятствия, а 0 – его отсутствие.

Роботу необходимо проанализировать данные и выполнить следующие задачи:

- 1) Робот должен двигаться по поверхности планеты, пока температура не поднимется выше 40°C или пока не встретит большое препятствие (обозначенное в массиве как 1). Определить, сколько участков робот пройдёт до остановки (переменная *count_before_stop*). Участок, на котором робот остановился, считать не нужно.
- 2) Робот должен проводить фотосъёмку каждого участка, где температура ниже 0°C, и определить, сколько снимков сделает робот до остановки (переменная *photo_count*). На участке, где остановился робот, можно провести фотосъёмку.
- 3) Робот должен подсчитать общее количество участков с нормальной температурой (от -35°C до +35°C) и без препятствий (переменная *count_normal_temp*).

Просуммируйте три числа, получившихся в результате выполнения указанных задач, и выберите из предложенных ниже вариантов ответа, чему равна эта сумма:

- A) 9
- Б) 10**
- B) 11
- Г) 12

Код программы:

```
#include <iostream>
```

```
using namespace std;
```

```
int main() {
```

```
    int temperatures[] = {-25, -40, 40, 0, -45, 30, -20};
```

```
    int obstacles[] = {0, 0, 0, 0, 0, 1, 1};
```

```
    int n = sizeof(temperatures) / sizeof(temperatures[0]);
```

```
    int i = 0;
```

```
    while (i < n && temperatures[i] <= 40 && obstacles[i] == 0) {
```

```
        i++;
```

```
    }
```

```
    int count_before_stop = i;
```

```
    int j = 0;
```

```
    int photo_count = 0;
```

```
    do {
```

```
        if (temperatures[j] < 0) {
```

```
            photo_count++;
```

```
        }
```

```
        j++;
```

```
    } while (j <= count_before_stop);
```

```
    int count_normal_temp = 0;
```

```
    for (int k = 0; k < n; k++) {
```

```
        if (temperatures[k] >= -35 && temperatures[k] <= 35 && obstacles[k] == 0) {
```

```
            count_normal_temp++;
```

```
        }
```

```

}

cout << "Количество участков до остановки: " << count_before_stop << endl;

cout << "Количество снимков: " << photo_count << endl;

cout << "Количество участков с нормальной температурой и без препятствий: " <<
count_normal_temp << endl;

return 0;

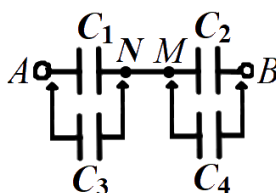
}

```

Ответ: Б) 10.

Задание 3

При проведении серии опытов на борту космической станции к клеммам **A-B** батареи конденсаторов C_1 и C_2 подключили источник Э.Д.С. (см. рисунок).



Когда конденсаторы C_1 и C_2 стали обладать суммарной энергией 222,0 мкДж, источник отключили и к батарее C_{12} подключили незаряженные конденсаторы C_3 и C_4 . Схема их подключения показана на рисунке. Ёмкости конденсаторов: $C_1 = 55,5$ мкФ, $C_3 = 24,5$ мкФ, $C_4 = 5,5$ мкФ. Площадь обкладок плоского конденсатора C_2 равна $6 \cdot 10^{-2}$ м², расстояние между ними $1,82 \cdot 10^{-6}$ м, диэлектрическая проницаемость вещества, заполняющего пространство между обкладками, $\epsilon = 84$. Во сколько раз максимальный заряд конденсатора C_3 оказался больше максимального заряда конденсатора C_4 ?

Приставка 1мк = 10^{-6} ; $\epsilon_0 = 8,85 \cdot 10^{-12}$ Ф/м.

Варианты ответа:

- А) 5,55
- Б) 3,40
- В) 2,92
- Г) 9,90
- Д) 1,67

Ответ: Д) 1,67

Задание 4

С поверхности неизвестной планеты вертикально вверх толкнули ядро с начальной скоростью 9 м/с. На высоту 9 м оно поднялось за 2 секунды. Учитывая результаты данного эксперимента, считая форму планеты шаром радиуса 100 км, вычислите радиус круговой

орбиты её спутника, движущегося со скоростью 300 м/с? Результат представьте в виде числа «М·10⁵ м» и выберите ответ с мантиссой «М».

Гравитационная постоянная $G = 6,67 \cdot 10^{-11} \text{ Н} \cdot \text{м}^2/\text{кг}^2$.

Возможные варианты ответа:

- А) 2
- Б) 3
- В) 4
- Г) 5**
- Д) 6
- Е) 7
- Ж) 8

Ответ: Г) 5

Задание 5

Некоторая планета совершает полный оборот вокруг своей звезды за 448 часов, а сутки (для наблюдателя на этой планете) длятся 56 часов. За сколько часов эта планета совершает один оборот вокруг своей оси, если направление вращения вокруг оси противоположно направлению вращения вокруг звезды? Выберите один вариант ответа из списка ниже с учётом округления.

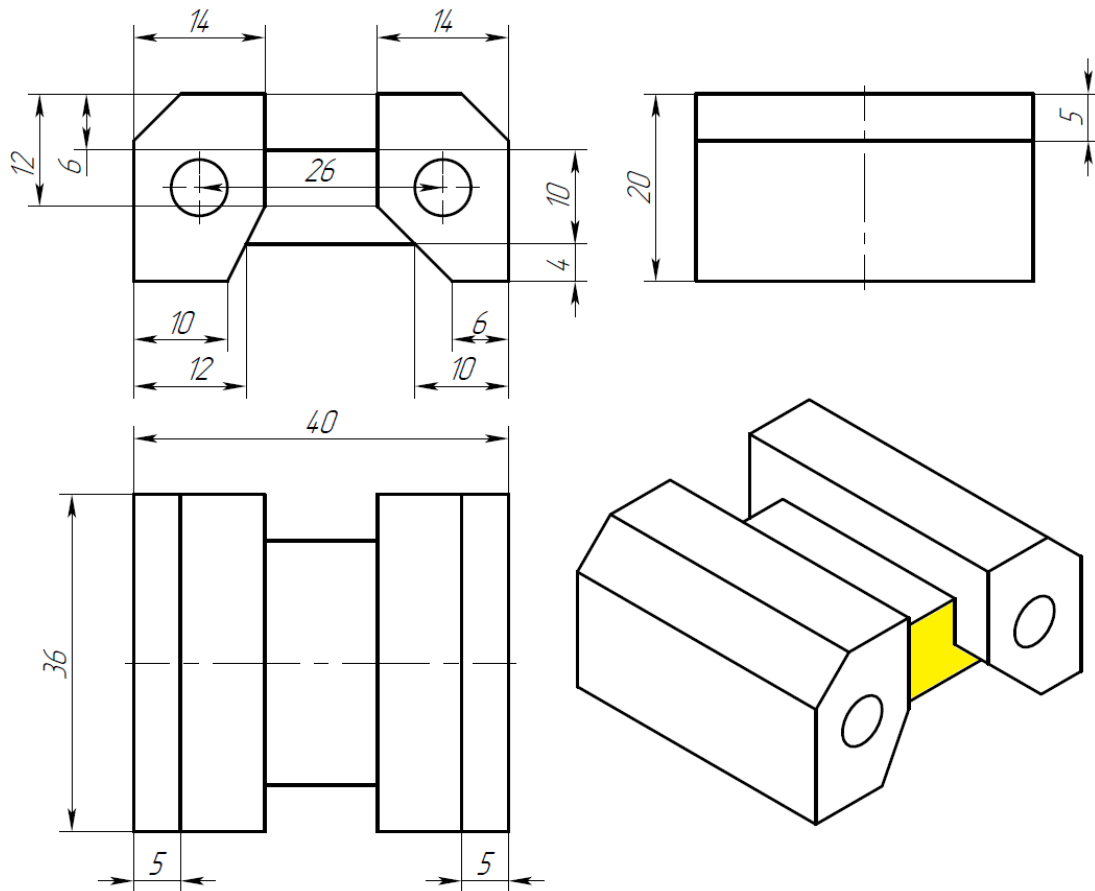
Варианты ответа:

- А) 49,78
- Б) 64,00**
- В) 56,00
- Г) 60,67

Ответ: Б) 64.

Задание 6

По размерам, имеющимся на чертеже, определите площадь выделенной цветом грани.



Варианты ответа:

- А) 128
- Б) 132**
- В) 130
- Г) 134

Ответ: Б) 132

Задание 7

Найти первую и вторую космические скорости для планеты Марс. Диаметр Марса – 6 770 км, ускорение свободного падения на поверхности составляет $3,711 \text{ м/с}^2$. Ответ выразите в км/с с одним знаком после запятой. При расчёте второй космической скорости необходимо воспользоваться полученным значением первой. В поля ответа необходимо ввести только числа.

Ответ: Первая космическая скорость: 3,5 км/с.

Вторая космическая скорость: 4,9 км/с.

Задание 8

Два тела движутся в одной плоскости. Если ввести на этой плоскости декартову систему координат, то траектории их движения совпадут с графиками функций $f(x) = x^2/18 - 1$ и $g(x) = \sin(\pi x - \pi/4)$. Определите количество точек пересечения этих траекторий.

Ответ: 12.

Задание 9

Космический крейсер оказался под атакой космических пиратов и отправил сигнал SOS. Два отряда космодесантников, находящихся на базе, получили сигнал и направились на помощь.

От базы (узел А) до крейсера (узел Д) можно добраться через сеть червоточин, соединяющих некоторые узлы в пространстве (планеты или точки в пространстве).

Космическое пространство можно представить в виде графа с пятью вершинами, где вершины – это узлы (планеты или точки в пространстве), а рёбра – это червоточины с определённой длиной. Ниже в матрице смежности дана информация о длине рёбер-червоточин.

	А	Б	В	Г	Д
А	0	4	5	7	0
Б	4	0	3	6	9
В	5	3	0	1	7
Г	7	6	1	0	8
Д	0	9	7	8	0

Отряд Альфа обязан начать движение с ребра с нечётной длиной и чередовать движение по рёбрам с нечётной и чётной длиной.

Отряд Бета может перемещаться только по рёбрам с нечётной длиной.

1) Какой из отрядов доберётся до крейсера быстрее? В случае равенства длин маршрутов выберите тот отряд, который прошёл через меньшее число узлов. В поле ответа введите название отряда одним словом с большой буквы без дополнительных символов (Альфа или Бета). **Ответ:** Бета.

2) Возможно ли, что один из отрядов не сможет добраться до крейсера, учитывая

ограничения на рёбра? Напишите в поле ответа знак «+», если это возможно, или знак «-» – в противном случае (без пробелов, кавычек или иных посторонних символов). **Ответ:-**

3) Введите в поле ответа длину кратчайшего маршрута (одним числом без пробелов и дополнительных знаков). **Ответ: 12.**

Задание 10

Космическое агентство разработало программу на языке C++ для управления запасами медицинских препаратов на борту исследовательского космического корабля. Программа использует три функции, работающие с локальными и глобальными переменными, а также реализующие рекурсию.

Функция `calculateMedicineConsumption` принимает количество членов экипажа и количество дней полёта, возвращает общее количество медицинских препаратов, которое израсходует экипаж.

Функция `checkMedicineLevel` принимает текущий уровень медицинских препаратов и минимально допустимый уровень. Если уровень препаратов опускается ниже 10% от начального запаса, выводится предупреждение.

Рекурсивная функция `calculateMedicineDaysLeft` принимает текущий запас медицинских препаратов, суточное потребление на одного члена экипажа и количество членов экипажа, возвращает количество дней, на которые хватит оставшегося запаса.

Входные данные:

- Начальный запас медицинских препаратов (переменная `medicineLevel`): 500 упаковок.
- Минимально допустимый уровень медицинских препаратов (переменная `minMedicineLevel`): 10% от начального запаса.
- Суточное потребление медицинских препаратов на одного члена экипажа (переменная `dailyMedicineConsumption`): 1 упаковка.
- Количество членов экипажа (переменная `crewMembers`): 8 человек.
- Количество дней полёта (переменная `flightDays`): 14 дней.

Вопросы:

1. Сколько медицинских препаратов израсходует экипаж за 14 дней полёта? **Ответ: 112.**
2. Какой будет уровень медицинских препаратов после 14 дней полёта? **Ответ: 388.**
3. На сколько дней хватит оставшегося запаса медицинских препаратов? **Ответ: 48.**

Код программы:

```
#include <iostream>
using namespace std;
```

```

int medicineLevel;

int calculateMedicineConsumption(int crewMembers, int days) {
    int medicineConsumed = crewMembers * days * 1;
    return medicineConsumed;
}

void checkMedicineLevel(int medicine, int minMedicineLevel) {
    if (medicine < minMedicineLevel) {
        cout << "Внимание: Уровень медицинских препаратов ниже допустимого!" << endl;
    }
}

int calculateMedicineDaysLeft(int medicine, int dailyMedicineConsumption, int crewMembers) {
    if (medicine < dailyMedicineConsumption * crewMembers) {
        return 0;
    }
    return 1 + calculateMedicineDaysLeft(medicine - dailyMedicineConsumption *
crewMembers, dailyMedicineConsumption, crewMembers);
}

int main() {
    medicineLevel = 500;
    int crewMembers = 8;
    int dailyMedicineConsumption = 1;
    int flightDays = 14;
    int minMedicineLevel = 0.1 * medicineLevel;

    int medicineConsumed = calculateMedicineConsumption(crewMembers, flightDays);
    cout << "Экипаж израсходует " << medicineConsumed << " упаковок медицинских
препаратов за " << flightDays << " дней" << endl;

    int remainingMedicine = medicineLevel - medicineConsumed;
    checkMedicineLevel(remainingMedicine, minMedicineLevel);
    cout << "Уровень медицинских препаратов после " << flightDays << " дней: " <<
remainingMedicine << " упаковок" << endl;

    int medicineDaysLeft = calculateMedicineDaysLeft(remainingMedicine,
dailyMedicineConsumption, crewMembers);
    cout << "Оставшегося запаса медицинских препаратов хватит на " <<
medicineDaysLeft << " дней" << endl;

    return 0;
}

```

Вариант 11

Задание 1

Жители планеты Марь для того, чтобы избежать сдвига привычных времён года, ввели понятие антивисокосного года, который на один день короче обычного. В обычном Марьском году ровно 400 Марьских дней. Антивисокосным является каждый 3-й год, а также каждый 13-й год. Если же очередной 3-й год совпадает с очередным 13-м, то убирается только один день. Найдите точную продолжительность года на планете Марь (выраженную в Марьских днях). Выберите из предложенных вариантов ответа, на сколько точная продолжительность года отличается от 400 дней.

Варианты ответа:

- А) 1/6
- Б) 17/49
- В) 5/13**
- Г) 39/400

Ответ: В) 5/13.

Задание 2

Космическое агентство разрабатывает автономного робота для исследования далёких планет. Робот должен уметь анализировать данные и выполнять повторяющиеся задачи в зависимости от полученных данных. Для этого разработана программа на языке C++, в которой используется несколько видов циклов и работа с массивами данных.

Входные данные:

- Робот получает данные о температуре на различных участках маршрута по поверхности планеты. Данные представлены в виде одномерного массива целых чисел (*temperatures[]*), где каждый элемент массива – это температура на конкретном участке.
- Робот фиксирует наличие препятствий на каждом участке, что представлено в виде другого массива (*obstacles[]*), где 1 означает наличие препятствия, а 0 – его отсутствие.

Роботу необходимо проанализировать данные и выполнить следующие задачи:

- 1) Робот должен двигаться по поверхности планеты, пока температура не поднимется выше 40°C или пока не встретит большое препятствие (обозначенное в массиве как 1). Определить, сколько участков робот пройдёт до остановки (переменная *count_before_stop*). Участок, на котором робот остановился, считать не нужно.
- 2) Робот должен проводить фотосъёмку каждого участка, где температура не выше 0°C, и определить, сколько снимков сделает робот до остановки (переменная *photo_count*). На участке, где остановился робот, можно провести фотосъёмку.
- 3) Робот должен подсчитать общее количество участков с нормальной температурой (от -35°C до +35°C) и без препятствий (переменная *count_normal_temp*).

Просуммируйте три числа, получившихся в результате выполнения указанных задач, и выберите из предложенных ниже вариантов ответа, чему равна эта сумма:

- A) 9
- Б) 10
- В) 11**
- Г) 12

Код программы:

```
#include <iostream>
```

```
using namespace std;
```

```
int main() {
```

```
    int temperatures[] = {-25, -40, 40, 0, -45, 30, -20};
```

```
    int obstacles[] = {0, 0, 0, 0, 0, 1, 1};
```

```
    int n = sizeof(temperatures) / sizeof(temperatures[0]);
```

```
    int i = 0;
```

```
    while (i < n && temperatures[i] <= 40 && obstacles[i] == 0) {
```

```
        i++;
```

```
    }
```

```
    int count_before_stop = i;
```

```
    int j = 0;
```

```
    int photo_count = 0;
```

```
    do {
```

```
        if (temperatures[j] <= 0) {
```

```
            photo_count++;
```

```
        }
```

```
        j++;
```

```
    } while (j <= count_before_stop);
```

```
    int count_normal_temp = 0;
```

```
    for (int k = 0; k < n; k++) {
```

```
        if (temperatures[k] >= -35 && temperatures[k] <= 35 && obstacles[k] == 0) {
```

```
            count_normal_temp++;
```

```
        }
```

```

}

cout << "Количество участков до остановки: " << count_before_stop << endl;

cout << "Количество снимков: " << photo_count << endl;

cout << "Количество участков с нормальной температурой и без препятствий: " <<
count_normal_temp << endl;

return 0;

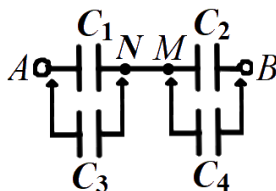
}

```

Ответ: В) 11.

Задание 3

При проведении серии опытов на борту космической станции к клеммам **A-B** батареи конденсаторов C_1 и C_2 подключили источник Э.Д.С. (см. рисунок).



Когда заряд конденсатора C_1 достиг $34,0$ мкКл, источник отключили и к батарее C_{12} подключили незаряженные конденсаторы C_3 и C_4 . Схема их подключения показана на рисунке. Ёмкости конденсаторов: $C_2 = 24,4$ мкФ, $C_3 = 6,0$ мкФ, $C_4 = 29,0$ мкФ. Площадь обкладок плоского конденсатора C_1 равна $4 \cdot 10^{-2}$ м², расстояние между ними $3,54 \cdot 10^{-6}$ м, диэлектрическая проницаемость вещества, заполняющего пространство между обкладками, $\epsilon = 63$. Рассчитайте разность потенциалов на клеммах **A-B** после максимальной зарядки конденсаторов C_3 и C_4 .

Приставка $1\text{мк} = 10^{-6}$; $\epsilon_0 = 8,85 \cdot 10^{-12}$ Ф/м.

Варианты ответа:

- А) 5,5 В
- Б) 3,4 В**
- В) 2,1 В
- Г) 0,9 В
- Д) 6,2 В

Ответ: Б) 3,4 В

Задание 4

С поверхности неизвестной планеты вертикально вверх толкнули ядро. За время подъёма ядра на максимальную высоту h спутник планеты преодолел путь 720 м, двигаясь по

круговой орбите на высоте 30 км над её поверхностью. Считая форму планеты шаром радиуса 110 км и учитывая параметры движения спутника, вычислите максимальную высоту h полёта ядра. Результат представьте в метрах.

Гравитационная постоянная $G = 6,67 \cdot 10^{-11} \text{ Н} \cdot \text{м}^2/\text{кг}^2$.

Возможные варианты ответа:

- А) 1 м
- Б) 2 м
- В) 3 м**
- Г) 4 м
- Д) 5 м
- Е) 6 м
- Ж) 7 м

Ответ: В) 3 м

Задание 5

Некоторая планета совершает полный оборот вокруг своей звезды за 660 часов, а сутки (для наблюдателя на этой планете) длятся 55 часов. За сколько часов эта планета совершает один оборот вокруг своей оси, если направление вращения вокруг оси противоположно направлению вращения вокруг звезды? Выберите один вариант ответа из списка ниже с учётом округления.

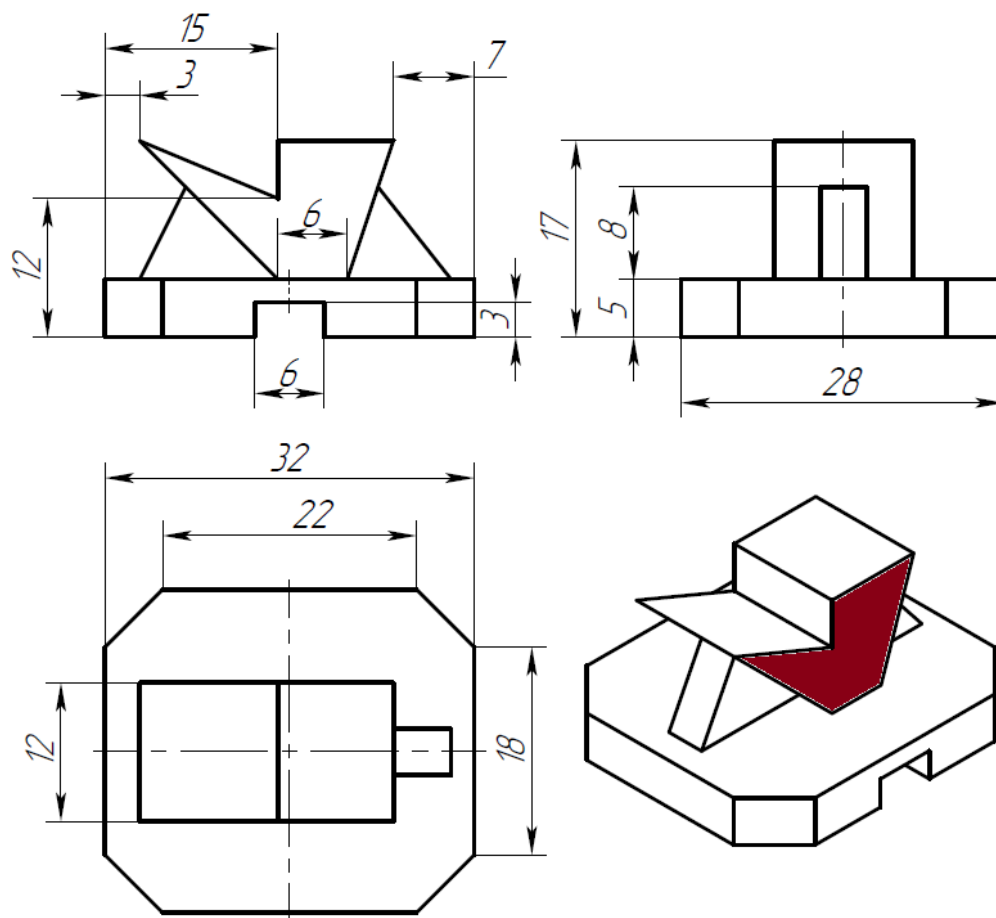
Варианты ответа:

- А) 50,77
- Б) 55,00
- В) 57,14
- Г) 60,00**

Ответ: Г) 60,00.

Задание 6

По размерам, имеющимся на чертеже, определите площадь выделенной цветом грани.



Варианты ответа:

- А) 132
- Б) 134
- В) 136
- Г) **138**

Ответ: Г) 138

Задание 7

Найти первую и вторую космические скорости для планеты Плутон. Диаметр Плутона – 6 000 км, ускорение свободного падения на поверхности составляет $0,617 \text{ м/с}^2$. Ответ выразите в км/с с одним знаком после запятой. При расчёте второй космической скорости необходимо воспользоваться полученным значением первой. В поля ответа необходимо ввести только числа.

Ответ: Первая космическая скорость: 1,4 км/с.

Вторая космическая скорость: 2,0 км/с.

Задание 8

Два тела движутся в одной плоскости. Если ввести на этой плоскости декартову систему координат, то траектории их движения совпадут с графиками функций $f(x) = \frac{(x-3)^2}{8} - 1$ и $g(x) = \sin(2\pi x)$. Определите количество точек пересечения этих траекторий.

Ответ: 16.

Задание 9

Космический крейсер оказался под атакой космических пиратов и отправил сигнал SOS. Два отряда космодесантников, находящихся на базе, получили сигнал и направились на помощь.

От базы (узел А) до крейсера (узел Д) можно добраться через сеть червоточин, соединяющих некоторые узлы в пространстве (планеты или точки в пространстве).

Космическое пространство можно представить в виде графа с пятью вершинами, где вершины – это узлы (планеты или точки в пространстве), а рёбра – это червоточины с определённой длиной. Ниже в матрице смежности дана информация о длине рёбер-червоточин.

	А	Б	В	Г	Д
А	0	6	8	7	0
Б	6	0	4	2	9
В	8	4	0	1	5
Г	7	2	1	0	9
Д	0	9	5	9	0

Отряд Альфа обязан начать движение с ребра с чётной длиной и чередовать движение по рёбрам с нечётной и чётной длиной.

Отряд Бета может перемещаться только по рёбрам с нечётной длиной.

1) Какой из отрядов доберётся до крейсера быстрее? В случае равенства длин маршрутов выберите тот отряд, который прошёл через меньшее число узлов. В поле ответа введите название отряда одним словом с большой буквы без дополнительных символов (Альфа или Бета). **Ответ:** Альфа.

2) Возможно ли, что один из отрядов не сможет добраться до крейсера, учитывая

ограничения на рёбра? Напишите в поле ответа знак «+», если это возможно, или знак «-» – в противном случае (без пробелов, кавычек или иных посторонних символов). **Ответ:** -.

3) Введите в поле ответа длину кратчайшего маршрута (одним числом без пробелов и дополнительных знаков). **Ответ:** 13.

Задание 10

Космическое агентство разработало программу на языке C++ для управления запасами топлива и мониторинга дистанции до цели на исследовательском космическом корабле. Программа использует три функции, работающие с локальными и глобальными переменными, а также реализующие рекурсию.

Функция `calculateFuelConsumption` принимает расстояние до цели и расход топлива на единицу расстояния, возвращает общее количество топлива, которое будет израсходовано.

Функция `checkFuelLevel` принимает текущий уровень топлива и минимально допустимый уровень. Если уровень топлива опускается ниже 25% от начального запаса, выводится предупреждение.

Рекурсивная функция `calculateDistanceDaysLeft` принимает оставшееся расстояние до цели и среднее расстояние, которое корабль проходит за день, и возвращает количество дней, необходимых для достижения цели.

Входные данные:

- Начальный запас топлива (переменная `fuelLevel`): 1000 тонн.
- Минимально допустимый уровень топлива (переменная `minFuelLevel`): 25% от начального запаса.
- Расход топлива на единицу расстояния (переменная `fuelConsumptionPerUnit`): 5 тонн.
- Расстояние до цели (переменная `distanceToTarget`): 150 единиц.
- Среднее расстояние, которое корабль проходит за день (переменная `dailyDistance`): 10 единиц.

Вопросы:

1. Сколько топлива будет израсходовано на расстояние до цели? **Ответ:** 750.
2. Какой будет уровень топлива после достижения цели? **Ответ:** 250.
3. Сколько дней потребуется для достижения цели? **Ответ:** 15.

Код программы:

```
#include <iostream>
using namespace std;

int fuelLevel;
```

```

int calculateFuelConsumption(int distance, int fuelConsumptionPerUnit) {
    int fuelConsumed = distance * fuelConsumptionPerUnit;
    return fuelConsumed;
}

void checkFuelLevel(int fuel, int minFuelLevel) {
    if (fuel < minFuelLevel) {
        cout << "Внимание: Уровень топлива ниже допустимого!" << endl;
    }
}

int calculateDistanceDaysLeft(int remainingDistance, int dailyDistance) {
    if (remainingDistance <= 0) {
        return 0;
    }
    return 1 + calculateDistanceDaysLeft(remainingDistance - dailyDistance, dailyDistance);
}

int main() {
    fuelLevel = 1000;
    int fuelConsumptionPerUnit = 5;
    int distanceToTarget = 150;
    int dailyDistance = 10;
    int minFuelLevel = 0.25 * fuelLevel;

    int fuelConsumed = calculateFuelConsumption(distanceToTarget, fuelConsumptionPerUnit);
    cout << "Топливо, необходимое для достижения цели: " << fuelConsumed << " тонн"
    << endl;

    int remainingFuel = fuelLevel - fuelConsumed;
    checkFuelLevel(remainingFuel, minFuelLevel);
    cout << "Уровень топлива после достижения цели: " << remainingFuel << " тонн" <<
    endl;

    int daysNeeded = calculateDistanceDaysLeft(distanceToTarget, dailyDistance);
    cout << "Потребуется " << daysNeeded << " дней для достижения цели" << endl;

    return 0;
}

```

Вариант 12

Задание 1

Жители планеты Марь для того, чтобы избежать сдвига привычных времён года, ввели понятие антивисокосного года, который на один день короче обычного. В обычном Марьском году ровно 140 Марьских дней. Антивисокосным является каждый 4-й год, а также каждый 7-й год. Если же очередной 4-й год совпадает с очередным 7-м, то убирается только один день. Найдите точную продолжительность года на планете Марь (выраженную в Марьских днях). Выберите из предложенных вариантов ответа, на сколько точная продолжительность года отличается от 140 дней.

Варианты ответа:

- А) $5/14$
- Б) $1/3$
- В) $11/2$
- Г) $3/35$

Ответ: А) $5/14$

Задание 2

Космическое агентство разрабатывает автономного робота для исследования далёких планет. Робот должен уметь анализировать данные и выполнять повторяющиеся задачи в зависимости от полученных данных. Для этого разработана программа на языке C++, в которой используется несколько видов циклов и работа с массивами данных.

Входные данные:

- Робот получает данные о температуре на различных участках маршрута по поверхности планеты. Данные представлены в виде одномерного массива целых чисел (*temperatures[]*), где каждый элемент массива – это температура на конкретном участке.
- Робот фиксирует наличие препятствий на каждом участке, что представлено в виде другого массива (*obstacles[]*), где 1 означает наличие препятствия, а 0 – его отсутствие.

Роботу необходимо проанализировать данные и выполнить следующие задачи:

- 1) Робот должен двигаться по поверхности планеты, пока температура не поднимется выше 40°C или пока не встретит большое препятствие (обозначенное в массиве как 1). Определить, сколько участков робот пройдёт до остановки (переменная *count_before_stop*). Участок, на котором робот остановился, считать не нужно.
- 2) Робот должен проводить фотосъёмку каждого участка, где температура не ниже 0°C, и определить, сколько снимков сделает робот до остановки (переменная *photo_count*). На участке, где остановился робот, можно провести фотосъёмку.
- 3) Робот должен подсчитать общее количество участков с нормальной температурой (от -35°C до +35°C) и без препятствий (переменная *count_normal_temp*).

Просуммируйте три числа, получившихся в результате выполнения указанных задач, и выберите из предложенных ниже вариантов ответа, чему равна эта сумма:

- A) 9
- Б) 10
- В) 11
- Г) 12

Код программы:

```
#include <iostream>
```

```
using namespace std;
```

```
int main() {
```

```
    int temperatures[] = {-25, -35, 35, 0, -45, 30, -20};
```

```
    int obstacles[] = {0, 0, 0, 0, 0, 1, 1};
```

```
    int n = sizeof(temperatures) / sizeof(temperatures[0]);
```

```
    int i = 0;
```

```
    while (i < n && temperatures[i] <= 40 && obstacles[i] == 0) {
```

```
        i++;
```

```
    }
```

```
    int count_before_stop = i;
```

```
    int j = 0;
```

```
    int photo_count = 0;
```

```
    do {
```

```
        if (temperatures[j] >= 0) {
```

```
            photo_count++;
```

```
        }
```

```
        j++;
```

```
    } while (j <= count_before_stop);
```

```
    int count_normal_temp = 0;
```

```
    for (int k = 0; k < n; k++) {
```

```
        if (temperatures[k] >= -35 && temperatures[k] <= 35 && obstacles[k] == 0) {
```

```
            count_normal_temp++;
```

```
        }
```

```

}

cout << "Количество участков до остановки: " << count_before_stop << endl;

cout << "Количество снимков: " << photo_count << endl;

cout << "Количество участков с нормальной температурой и без препятствий: " <<
count_normal_temp << endl;

return 0;

}

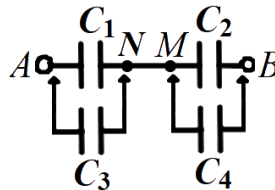
```

Ответ: Г) 12.

Задание 3

При проведении серии опытов на борту космической станции к клеммам **A-B** батареи незаряженных конденсаторов C_1 и C_2 подключили источник Э.Д.С. (см. рисунок).

Когда батарея C_{12} зарядилась, источник отключили и к батарее C_{12} подключили



незаряженные конденсаторы C_3 и C_4 . Схема их подключения показана на рисунке. Энергия конденсатора C_4 после его подключения к конденсатору C_2 достигла максимального значения $1,04$ мкДж. Ёмкости конденсаторов $C_1 = 3,25$ мкФ, $C_3 = 16,75$ мкФ, $C_4 = 2,40$ мкФ. Площадь обкладок плоского конденсатора C_2 равна $6,5 \cdot 10^{-2}$ м², расстояние между ними $1,77 \cdot 10^{-6}$ м, диэлектрическая проницаемость вещества, заполняющего пространство между обкладками, $\epsilon = 40$. Рассчитайте максимальный заряд конденсатора C_3 .

Приставка $1\text{мк} = 10^{-6}$; $\epsilon_0 = 8,85 \cdot 10^{-12}$ Ф/м.

Варианты ответа:

- А) 12 мкКл
- Б) 4 мкКл
- В) 2 мкКл
- Г) 9 мкКл
- Д) 16 мкКл

Ответ: А). 12 мкКл

Задание 4

С поверхности неизвестной планеты вертикально вверх толкнули ядро. За время подъёма ядра на максимальную высоту 4,5 м спутник этой планеты преодолел путь 900 м, двигаясь по круговой орбите. Учитывая параметры движения ядра и спутника, считая форму планеты шаром радиуса 120 км, рассчитайте высоту орбиты спутника над поверхностью планеты. Результат представьте в виде числа «М·10⁴ м» и выберите ответ с мантиссой «М».

Гравитационная постоянная $G = 6,67 \cdot 10^{-11} \text{ Н} \cdot \text{м}^2 / \text{кг}^2$.

Возможные варианты ответа:

- А) 3
- Б) 4**
- В) 5
- Г) 6
- Д) 7
- Е) 8
- Ж) 9

Ответ: Б) 4

Задание 5

Некоторая планета совершает полный оборот вокруг своей звезды за 720 часов, а сутки (для наблюдателя на этой планете) длятся 60 часов. За сколько часов эта планета совершает один оборот вокруг своей оси, если направление вращения вокруг оси противоположно направлению вращения вокруг звезды? Выберите один вариант ответа из списка ниже с учётом округления.

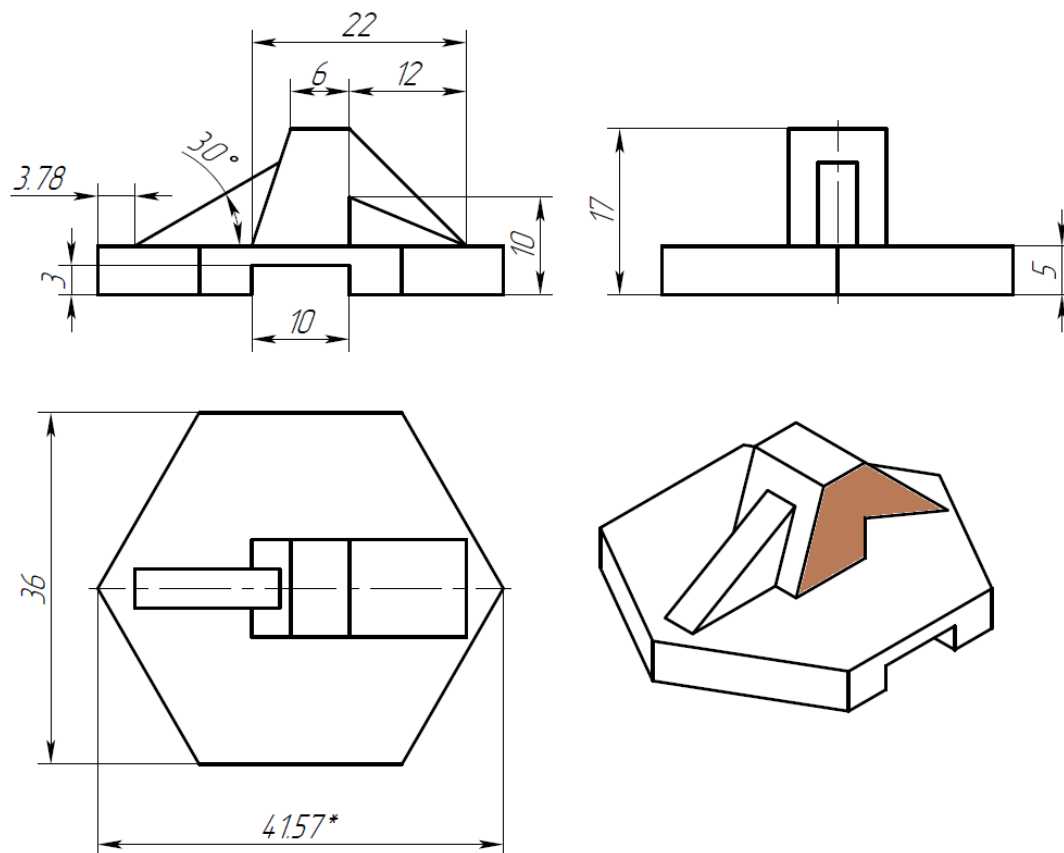
Варианты ответа:

- А) 65,45**
- Б) 60,00
- В) 55,38
- Г) 68,57

Ответ: А) 65,45.

Задание 6

По размерам, имеющимся на чертеже, определите площадь выделенной цветом грани.



Варианты ответа:

- А) 132
- Б) 134
- В) 136
- Г) 138

Ответ: Г) 138

Задание 7

Найти первую и вторую космические скорости для планеты Сатурн. Диаметр Сатурна – 116 820 км, ускорение свободного падения на поверхности составляет $10,44 \text{ м/с}^2$. Ответ выразите в км/с с одним знаком после запятой. При расчёте второй космической скорости необходимо воспользоваться полученным значением первой. В поля ответа необходимо ввести только числа.

Ответ: Первая космическая скорость: 24,7 км/с.

Вторая космическая скорость: 34,9 км/с.

Задание 8

Два тела движутся в одной плоскости. Если ввести на этой плоскости декартову систему координат, то траектории их движения совпадут с графиками функций $f(x) = \frac{(x-3)^2}{19} - 1$ и $g(x) = \sin(\pi x)$. Определите количество точек пересечения этих траекторий.

Ответ: 12.

Задание 9

Космический крейсер оказался под атакой космических пиратов и отправил сигнал SOS. Два отряда космодесантников, находящихся на базе, получили сигнал и направились на помощь.

От базы (узел А) до крейсера (узел Д) можно добраться через сеть червоточин, соединяющих некоторые узлы в пространстве (планеты или точки в пространстве).

Космическое пространство можно представить в виде графа с пятью вершинами, где вершины – это узлы (планеты или точки в пространстве), а рёбра – это червоточины с определённой длиной. Ниже в матрице смежности дана информация о длине рёбер-червоточин.

	А	Б	В	Г	Д
А	0	4	8	6	0
Б	4	0	5	2	3
В	8	5	0	1	4
Г	6	2	1	0	5
Д	0	3	4	5	0

Отряд Альфа обязан начать движение с ребра с чётной длиной и чередовать движение по рёбрам с нечётной и чётной длиной.

Отряд Бета может перемещаться только по рёбрам с нечётной длиной.

1) Какой из отрядов доберётся до крейсера быстрее? В случае равенства длин маршрутов выберите тот отряд, который прошёл через меньшее число узлов. В поле ответа введите название отряда одним словом с большой буквы без дополнительных символов (Альфа или Бета). **Ответ:** Альфа.

2) Возможно ли, что один из отрядов не сможет добраться до крейсера, учитывая

ограничения на рёбра? Напишите в поле ответа знак «+», если это возможно, или знак «-» – в противном случае (без пробелов, кавычек или иных посторонних символов). **Ответ:** +.

3) Введите в поле ответа длину кратчайшего маршрута (одним числом без пробелов и дополнительных знаков). **Ответ:** 7.

Задание 10

Космическое агентство разработало программу на языке C++ для управления энергетическими ресурсами и мониторинга состояния батарей на борту исследовательского космического корабля. Программа использует три функции, работающие с локальными и глобальными переменными, а также реализующие рекурсию.

Функция `calculateEnergyGeneration` принимает количество солнечных панелей, количество дней полёта и выработку энергии одной панелью в день, возвращает общее количество энергии, которое будет сгенерировано.

Функция `checkBatteryLevel` принимает текущий уровень заряда батарей и максимальный уровень. Если уровень заряда батарей опускается ниже 20% от максимального значения, выводится предупреждение.

Рекурсивная функция `calculateDaysUntilBatteryDepletion` принимает текущий уровень заряда батарей и суточное потребление энергии, возвращает количество дней, на которые хватит оставшегося заряда.

Входные данные:

- Начальный уровень заряда батарей (переменная `batteryLevel`): 5000 единиц.
- Максимальный уровень заряда батарей (переменная `maxBatteryLevel`): 5000 единиц.
- Суточное потребление энергии (переменная `dailyEnergyConsumption`): 400 единиц.
- Количество солнечных панелей (переменная `solarPanels`): 5 штук.
- Выработка энергии одной панелью в день (переменная `energyPerPanelPerDay`): 50 единиц.
- Количество дней полёта (переменная `flightDays`): 8 дней.

Вопросы:

1. Сколько энергии будет сгенерировано за 8 дней полёта? **Ответ:** 2000.
2. Какой будет уровень заряда батарей после 8 дней полёта? **Ответ:** 3800.
3. На сколько дней хватит оставшегося заряда батарей при текущем потреблении энергии? **Ответ:** 9.

Код программы:

```
#include <iostream>
using namespace std;
```

```

int batteryLevel;

int calculateEnergyGeneration(int solarPanels, int days, int energyPerPanelPerDay) {
    int energyGenerated = solarPanels * days * energyPerPanelPerDay;
    return energyGenerated;
}

void checkBatteryLevel(int battery, int maxBatteryLevel) {
    if (battery < 0.2 * maxBatteryLevel) {
        cout << "Внимание: Уровень заряда батарей ниже 20%!" << endl;
    }
}

int calculateDaysUntilBatteryDepletion(int battery, int dailyEnergyConsumption) {
    if (battery < dailyEnergyConsumption) {
        return 0;
    }
    return 1 + calculateDaysUntilBatteryDepletion(battery - dailyEnergyConsumption,
dailyEnergyConsumption);
}

int main() {
    batteryLevel = 5000;
    int maxBatteryLevel = 5000;
    int dailyEnergyConsumption = 400;
    int solarPanels = 5;
    int energyPerPanelPerDay = 50;
    int flightDays = 8;

    int energyGenerated = calculateEnergyGeneration(solarPanels, flightDays,
energyPerPanelPerDay);
    cout << "Сгенерировано энергии за " << flightDays << " дней: " << energyGenerated <<
" единиц" << endl;

    batteryLevel = batteryLevel - (dailyEnergyConsumption * flightDays) + energyGenerated;
    checkBatteryLevel(batteryLevel, maxBatteryLevel);
    cout << "Уровень заряда батарей после " << flightDays << " дней: " << batteryLevel <<
" единиц" << endl;

    int daysUntilDepletion = calculateDaysUntilBatteryDepletion(batteryLevel,
dailyEnergyConsumption);
    cout << "Оставшегося заряда батарей хватит на " << daysUntilDepletion << " дней"
<< endl;

    return 0;
}

```

Вариант 13

Задание 1

Жители планеты Линь для того, чтобы избежать сдвига привычных времён года, ввели понятие антивисокосного года, который на один день короче обычного. В обычном Линьском году ровно 240 Линьских дней. Антивисокосным является каждый 4-й год, а также каждый 5-й год. Если же очередной 4-й год совпадает с очередным 5-м, то год считается обычным. Найдите точную продолжительность года на планете Линь (выраженную в Линьских днях). Выберите из предложенных вариантов ответа, на сколько точная продолжительность года отличается от 240 дней.

Варианты ответа:

- А) 1/2
- Б) 7/20**
- В) 9/20
- Г) 1/12

Ответ: Б) 7/20.

Задание 2

Космическое агентство разрабатывает автономного робота для исследования далёких планет. Робот должен уметь анализировать данные и выполнять повторяющиеся задачи в зависимости от полученных данных. Для этого разработана программа на языке C++, в которой используется несколько видов циклов и работа с массивами данных.

Входные данные:

- Робот получает данные о температуре на различных участках маршрута по поверхности планеты. Данные представлены в виде одномерного массива целых чисел (*temperatures[]*), где каждый элемент массива – это температура на конкретном участке.
- Робот фиксирует наличие препятствий на каждом участке, что представлено в виде другого массива (*obstacles[]*), где 1 означает наличие препятствия, а 0 – его отсутствие.

Роботу необходимо проанализировать данные и выполнить следующие задачи:

- 1) Робот должен двигаться по поверхности планеты, пока температура не поднимется выше 40°C или пока не встретит большое препятствие (обозначенное в массиве как 1). Определить, сколько участков робот пройдёт до остановки (переменная *count_before_stop*). Участок, на котором робот остановился, считать не нужно.
- 2) Робот должен проводить фотосъёмку каждого участка, где температура ниже 0°C, и определить, сколько снимков сделает робот до остановки (переменная *photo_count*). На участке, где остановился робот, можно провести фотосъёмку.
- 3) Робот должен подсчитать общее количество участков с нормальной температурой (от -35°C до +35°C) и без препятствий (переменная *count_normal_temp*).

Просуммируйте три числа, получившихся в результате выполнения указанных задач, и выберите из предложенных ниже вариантов ответа, чему равна эта сумма:

- A) 6
- Б) 7
- В) 8
- Г) 9

Код программы:

```
#include <iostream>
```

```
using namespace std;
```

```
int main() {
```

```
    int temperatures[] = {-25, -40, 45, 0, 40, 30, -20};
```

```
    int obstacles[] = {0, 0, 0, 0, 0, 1, 1};
```

```
    int n = sizeof(temperatures) / sizeof(temperatures[0]);
```

```
    int i = 0;
```

```
    while (i < n && temperatures[i] <= 40 && obstacles[i] == 0) {
```

```
        i++;
```

```
    }
```

```
    int count_before_stop = i;
```

```
    int j = 0;
```

```
    int photo_count = 0;
```

```
    do {
```

```
        if (temperatures[j] < 0) {
```

```
            photo_count++;
```

```
        }
```

```
        j++;
```

```
    } while (j <= count_before_stop);
```

```
    int count_normal_temp = 0;
```

```
    for (int k = 0; k < n; k++) {
```

```
        if (temperatures[k] >= -35 && temperatures[k] <= 35 && obstacles[k] == 0) {
```

```
            count_normal_temp++;
```

```
        }
```

```

}

cout << "Количество участков до остановки: " << count_before_stop << endl;

cout << "Количество снимков: " << photo_count << endl;

cout << "Количество участков с нормальной температурой и без препятствий: " <<
count_normal_temp << endl;

return 0;

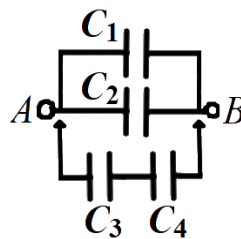
}

```

Ответ: А) 6.

Задание 3

При проведении серии опытов на борту космической станции батареею конденсаторов C_1 и C_2 зарядили, подключив к клеммам **A-B** источник Э.Д.С. (см. рисунок).



Затем источник отключили и к клеммам **A-B** подключили батарею незаряженных конденсаторов C_3 и C_4 . Вскоре энергия конденсатора C_4 достигла максимального значения 6,25 мкДж. Ёмкости конденсаторов: $C_1 = 0,67$ мкФ, $C_3 = 2,0$ мкФ, $C_4 = 8,0$ мкФ. Площадь обкладок плоского конденсатора C_2 равна $7,0 \cdot 10^{-2}$ м², расстояние между ними $1,3 \cdot 10^{-5}$ м, диэлектрическая проницаемость вещества, заполняющего пространство между обкладками, $\epsilon = 42$. Рассчитайте заряд конденсатора C_2 перед подключением блока C_{34} .

Приставка 1мк = 10^{-6} ; $\epsilon_0 = 8,85 \cdot 10^{-12}$ Ф/м.

Варианты ответа:

- А) 12 мкКл
- Б) 42 мкКл
- В) 50 мкКл
- Г) **20 мкКл**
- Д) 36 мкКл

Ответ: Г) 20 мкКл.

Задание 4

С поверхности неизвестной планеты вверх под углом 60° к горизонту толкнули ядро с начальной скоростью 8 м/с . Через 12 секунд скорость ядра уменьшилась вдвое. Используя результаты данного эксперимента, считая форму планеты шаром радиуса 125 км , рассчитайте скорость её спутника на круговой орбите. Высота орбиты 125 км над поверхностью планеты. Результат представьте в виде числа « $M \cdot 10^2 \text{ м/с}$ » и выберите ответ с мантиссой « M ».

Гравитационная постоянная $G = 6,67 \cdot 10^{-11} \text{ Н} \cdot \text{м}^2/\text{кг}^2$.

Возможные варианты ответа:

- А) 8,5
- Б) 7,8
- В) 5,3
- Г) 4,0
- Д) 2,9
- Е) 6,2
- Ж) 1,9**

Ответ: Ж) 1,9.

Задание 5

Некоторая планета совершает полный оборот вокруг своей звезды за 720 часов, а сутки (для наблюдателя на этой планете) длятся 45 часов. За сколько часов эта планета совершает один оборот вокруг своей оси, если направление вращения вокруг оси противоположно направлению вращения вокруг звезды? Выберите один вариант ответа из списка ниже с учётом округления.

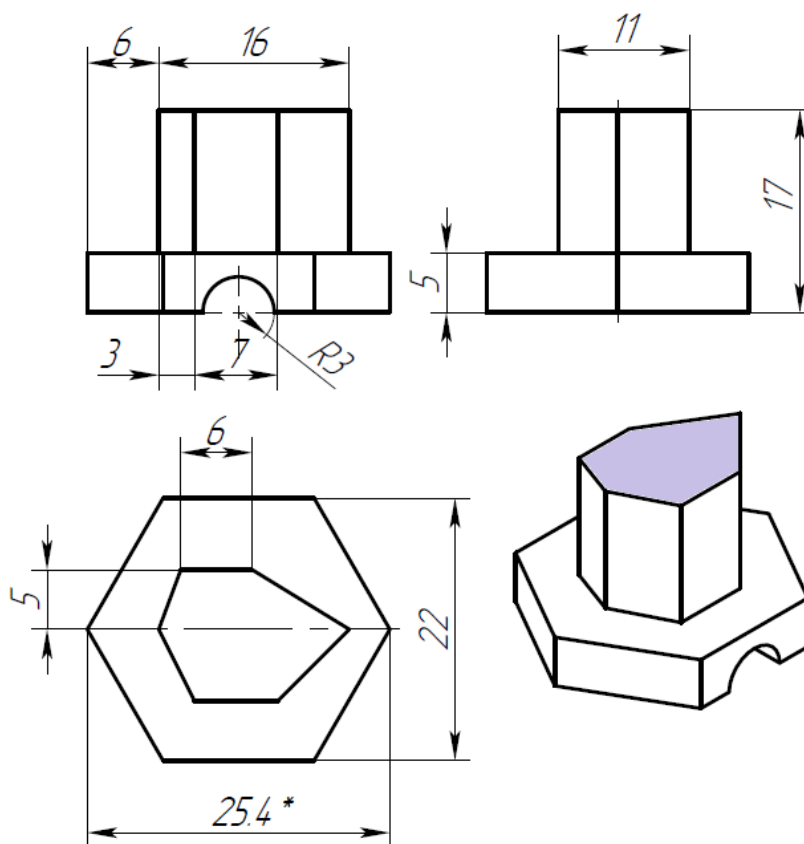
Варианты ответа:

- А) 45,00
- Б) 48,00**
- В) 55,38
- Г) 46,45

Ответ: Б) 48,00.

Задание 6

По размерам, имеющимся на чертеже, определите площадь выделенной цветом грани.



Варианты ответа:

- А) 120
- Б) 124**
- В) 128
- Г) 132

Ответ: Б) 124

Задание 7

Найти первую и вторую космические скорости для планеты Меркурий. Масса Меркурия составляет $3,3 \cdot 10^{23}$ кг, диаметр – 4 840 км. При расчётах принять гравитационную постоянную равной $6,67 \cdot 10^{-11}$ Н·м²/кг². Ответ выразите в км/с с одним знаком после запятой. При расчёте второй космической скорости необходимо воспользоваться полученным значением первой. В поля ответа необходимо ввести только числа.

Ответ: Первая космическая скорость: 3,0 км/с.

Вторая космическая скорость: 4,2 км/с.

Задание 8

Два тела движутся в одной плоскости. Если ввести на этой плоскости декартову систему координат, то траектории их движения совпадут с графиками функций $f(x) = x^2/14 - 1$ и $g(x) = \cos(\pi x - \pi/4)$. Определите количество точек пересечения этих траекторий.

Ответ: 10.

Задание 9

Космический крейсер оказался под атакой космических пиратов и отправил сигнал SOS. Два отряда космодесантников, находящихся на базе, получили сигнал и направились на помощь.

От базы (узел А) до крейсера (узел Д) можно добраться через сеть червоточин, соединяющих некоторые узлы в пространстве (планеты или точки в пространстве).

Космическое пространство можно представить в виде графа с пятью вершинами, где вершины – это узлы (планеты или точки в пространстве), а рёбра – это червоточины с определённой длиной. Ниже в матрице смежности дана информация о длине рёбер-червоточин.

	А	Б	В	Г	Д
А	0	4	7	2	0
Б	4	0	5	1	8
В	7	5	0	1	3
Г	2	1	1	0	9
Д	0	8	3	9	0

Отряд Альфа может перемещаться только по рёбрам с чётной длиной.

Отряд Бета обязан начать движение с ребра с нечётной длиной и чередовать движение по рёбрам с чётной и нечётной длиной.

1) Какой из отрядов доберётся до крейсера быстрее? В случае равенства длин маршрутов выберите тот отряд, который прошёл через меньшее число узлов. В поле ответа введите название отряда одним словом с большой буквы без дополнительных символов (Альфа или Бета). **Ответ:** Альфа.

2) Возможно ли, что один из отрядов не сможет добраться до крейсера, учитывая

ограничения на рёбра? Напишите в поле ответа знак «+», если это возможно, или знак «-» – в противном случае (без пробелов, кавычек или иных посторонних символов). **Ответ:** +.

3) Введите в поле ответа длину кратчайшего маршрута (одним числом без пробелов и дополнительных знаков). **Ответ:** 12.

Задание 10

Космическое агентство разработало программу на языке C++ для управления отходами и переработкой ресурсов на борту исследовательского космического корабля. Программа использует три функции, работающие с локальными и глобальными переменными, а также реализующие рекурсию.

Функция `calculateWasteProduction` принимает количество членов экипажа и количество дней полёта и возвращает общее количество отходов, которое будет произведено.

Функция `processWaste` принимает количество произведённых отходов и эффективность системы переработки (в процентах). Она обновляет глобальную переменную `wasteLevel` после переработки. Если уровень отходов превышает 80% от максимальной вместимости, выводится предупреждение.

Рекурсивная функция `calculateDaysUntilCapacity` принимает текущий уровень отходов, ежедневное чистое производство отходов и возвращает количество дней до заполнения системы отходов.

Входные данные:

- Начальный уровень отходов (переменная `wasteLevel`): 50 кг.
- Максимальная вместимость системы отходов (переменная `maxWasteCapacity`): 500 кг.
- Суточное производство отходов на одного члена экипажа (переменная `dailyWasteProduction`): 2 кг.
- Количество членов экипажа (переменная `crewMembers`): 5 человек.
- Количество дней полёта (переменная `flightDays`): 10 дней.
- Эффективность системы переработки (переменная `recyclingEfficiency`): 60%.

Вопросы:

1. Сколько отходов произведёт экипаж за 10 дней полёта? **Ответ:** 100.
2. Какой будет уровень отходов после переработки? **Ответ:** 90.
3. Через сколько дней система отходов достигнет максимальной вместимости?

Ответ: 103.

Код программы:

```
#include <iostream>
using namespace std;
```

```

int wasteLevel;

int calculateWasteProduction(int crewMembers, int days) {
    int wasteProduced = crewMembers * days * 2;
    return wasteProduced;
}

void processWaste(int waste, int recyclingEfficiency) {
    int recycledWaste = waste * recyclingEfficiency / 100;
    wasteLevel += waste - recycledWaste;
    if (wasteLevel > 0.8 * 500) {
        cout << "Внимание: Уровень отходов превышает 80% от максимальной
вместимости!" << endl;
    }
}

int calculateDaysUntilCapacity(int currentWaste, int netDailyWasteProduction, int
maxWasteCapacity) {
    if (currentWaste >= maxWasteCapacity) {
        return 0;
    }
    return 1 + calculateDaysUntilCapacity(currentWaste + netDailyWasteProduction,
netDailyWasteProduction, maxWasteCapacity);
}

int main() {
    wasteLevel = 50;
    int maxWasteCapacity = 500;
    int crewMembers = 5;
    int dailyWasteProduction = 2;
    int flightDays = 10;
    int recyclingEfficiency = 60;

    int wasteProduced = calculateWasteProduction(crewMembers, flightDays);
    cout << "Экипаж произведёт " << wasteProduced << " кг отходов за " << flightDays
<< " дней" << endl;

    processWaste(wasteProduced, recyclingEfficiency);
    cout << "Уровень отходов после переработки: " << wasteLevel << " кг" << endl;

    int dailyTotalWasteProduction = crewMembers * dailyWasteProduction;
    int netDailyWasteProduction = dailyTotalWasteProduction * (100 - recyclingEfficiency) /
100;

    int daysUntilCapacity = calculateDaysUntilCapacity(wasteLevel, netDailyWasteProduction,
maxWasteCapacity);
    cout << "Система отходов достигнет максимальной вместимости через " <<
daysUntilCapacity << " дней" << endl;

    return 0;
}

```

Вариант 14

Задание 1

Жители планеты Линь для того, чтобы избежать сдвига привычных времён года, ввели понятие антивисокосного года, который на один день короче обычного. В обычном Линьском году ровно 350 Линьских дней. Антивисокосным является каждый 5-й год, а также каждый 9-й год. Если же очередной 5-й год совпадает с очередным 9-м, то год считается обычным. Найдите точную продолжительность года на планете Линь (выраженную в Линьских днях). Выберите из предложенных вариантов ответа, на сколько точная продолжительность года отличается от 350 дней.

Варианты ответа:

- А) 2/45
- Б) 1/3
- В) 4/15**
- Г) 3/70

Ответ: В) 4/15.

Задание 2

Космическое агентство разрабатывает автономного робота для исследования далёких планет. Робот должен уметь анализировать данные и выполнять повторяющиеся задачи в зависимости от полученных данных. Для этого разработана программа на языке C++, в которой используется несколько видов циклов и работа с массивами данных.

Входные данные:

- Робот получает данные о температуре на различных участках маршрута по поверхности планеты. Данные представлены в виде одномерного массива целых чисел (*temperatures[]*), где каждый элемент массива – это температура на конкретном участке.
- Робот фиксирует наличие препятствий на каждом участке, что представлено в виде другого массива (*obstacles[]*), где 1 означает наличие препятствия, а 0 – его отсутствие.

Роботу необходимо проанализировать данные и выполнить следующие задачи:

- 1) Робот должен двигаться по поверхности планеты, пока температура не поднимется выше 40°C или пока не встретит большое препятствие (обозначенное в массиве как 1). Определить, сколько участков робот пройдёт до остановки (переменная *count_before_stop*). Участок, на котором робот остановился, считать не нужно.
- 2) Робот должен проводить фотосъёмку каждого участка, где температура не ниже 0°C, и определить, сколько снимков сделает робот до остановки (переменная *photo_count*). На участке, где остановился робот, можно провести фотосъёмку.
- 3) Робот должен подсчитать общее количество участков с нормальной температурой (от -35°C до +35°C) и без препятствий (переменная *count_normal_temp*).

Просуммируйте три числа, получившихся в результате выполнения указанных задач, и выберите из предложенных ниже вариантов ответа, чему равна эта сумма:

- A) 6
- Б) 7**
- B) 8
- Г) 9

Код программы:

```
#include <iostream>
```

```
using namespace std;
```

```
int main() {
```

```
    int temperatures[] = {-25, -40, 35, 45, 40, 0, -20};
```

```
    int obstacles[] = {0, 0, 0, 0, 0, 1, 1};
```

```
    int n = sizeof(temperatures) / sizeof(temperatures[0]);
```

```
    int i = 0;
```

```
    while (i < n && temperatures[i] <= 40 && obstacles[i] == 0) {
```

```
        i++;
```

```
    }
```

```
    int count_before_stop = i;
```

```
    int j = 0;
```

```
    int photo_count = 0;
```

```
    do {
```

```
        if (temperatures[j] <= 0) {
```

```
            photo_count++;
```

```
        }
```

```
        j++;
```

```
    } while (j <= count_before_stop);
```

```
    int count_normal_temp = 0;
```

```
    for (int k = 0; k < n; k++) {
```

```
        if (temperatures[k] >= -35 && temperatures[k] <= 35 && obstacles[k] == 0) {
```

```
            count_normal_temp++;
```

```
        }
```

```

}

cout << "Количество участков до остановки: " << count_before_stop << endl;

cout << "Количество снимков: " << photo_count << endl;

cout << "Количество участков с нормальной температурой и без препятствий: " <<
count_normal_temp << endl;

return 0;

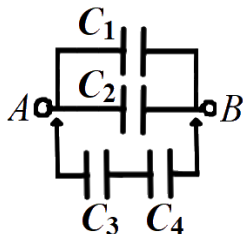
}

```

Ответ: Б) 7.

Задание 3

При проведении серии опытов на борту космической станции к клеммам **A-B** батареи конденсаторов C_1 и C_2 подключили источник Э.Д.С. (см. рисунок).



Когда батарея C_{12} зарядилась, источник отключили и к клеммам **A-B** подключили батарею незаряженных конденсаторов C_3 и C_4 . Вскоре заряд конденсатора C_3 достиг максимального значения, при котором разность потенциалов на его обкладках оказалась равной 2,88 В. Ёмкости конденсаторов: $C_2 = 0,6$ мкФ, $C_3 = 2,0$ мкФ, $C_4 = 0,5$ мкФ. Площадь обкладок плоского конденсатора C_1 равна $8,18 \cdot 10^{-2}$ м², расстояние между ними $1,52 \cdot 10^{-5}$ м, диэлектрическая проницаемость вещества, заполняющего пространство между обкладками, $\epsilon = 21$. Рассчитайте, во сколько раз уменьшился заряд конденсатора C_1 после максимальной зарядки конденсаторов C_3 и C_4 .

Приставка 1мк = 10^{-6} ; $\epsilon_0 = 8,85 \cdot 10^{-12}$ Ф/м.

Варианты ответа:

- А) 1,25
- Б) 3,14
- В) 2,15
- Г) 4,39
- Д) 8,35

Ответ: А) 1,25

Задание 4

С горизонтального участка поверхности неизвестной планеты под углом 45° толкнули ядро с начальной скоростью $14,14$ м/с. Через 20 секунд оно упало на горизонтальный участок поверхности. Используя результаты эксперимента, считая форму планеты шаром радиуса 90 км, вычислите для неё величину первой космической скорости. Результат представьте в виде числа « $M \cdot 10^2$ м/с» и выберите ответ с мантиссой «М».

Гравитационная постоянная $G = 6,67 \cdot 10^{-11} \text{ Н} \cdot \text{м}^2/\text{кг}^2$.

Возможные варианты ответа:

- А) 1
- Б) 2
- В) 3**
- Г) 4
- Д) 5
- Е) 6
- Ж) 7

Ответ: В) 3.

Задание 5

Некоторая планета совершает полный оборот вокруг своей звезды за 768 часов, а сутки (для наблюдателя на этой планете) длятся 48 часов. За сколько часов эта планета совершает один оборот вокруг своей оси, если направление вращения вокруг оси противоположно направлению вращения вокруг звезды? Выберите один вариант ответа из списка ниже с учётом округления.

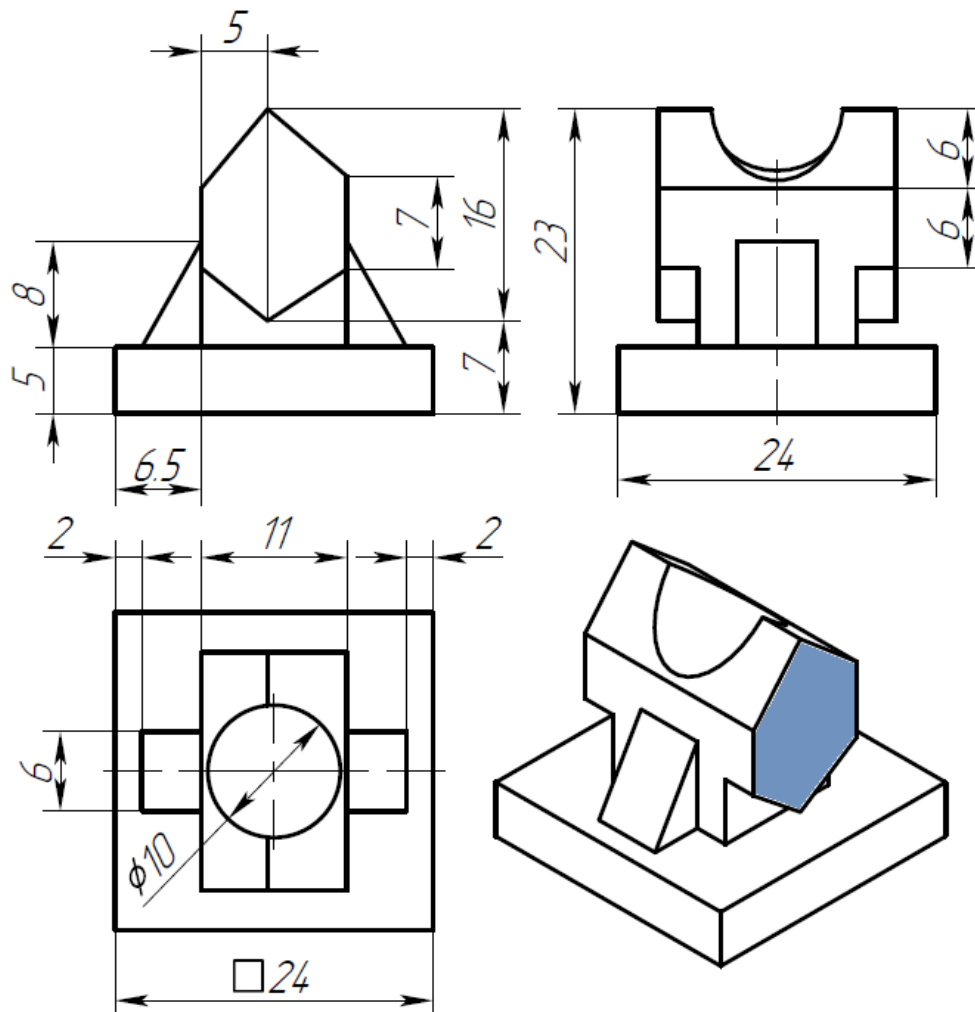
Варианты ответа:

- А) $54,86$
- Б) $48,00$
- В) $51,20$**
- Г) $45,17$

Ответ: В) $51,20$.

Задание 6

По размерам, имеющимся на чертеже, определите площадь выделенной цветом грани.



Варианты ответа:

- А) 120
- Б) 124**
- В) 128
- Г) 132

Ответ: Б) 124

Задание 7

Найти первую и вторую космические скорости для планеты Земля. Масса Земли составляет $6 \cdot 10^{24}$ кг, диаметр – 12 740 км. При расчётах принять гравитационную постоянную равной $6,67 \cdot 10^{-11}$ Н·м²/кг². Ответ выразите в км/с с одним знаком после запятой. При расчёте второй космической скорости необходимо воспользоваться полученным значением первой. В поля ответа необходимо ввести только числа.

Ответ: Первая космическая скорость: 7,9 км/с.

Вторая космическая скорость: 11,2 км/с.

Задание 8

Два тела движутся в одной плоскости. Если ввести на этой плоскости декартову систему координат, то траектории их движения совпадут с графиками функций $f(x) = x^2/10 - 1$ и $g(x) = \cos(2\pi x + \pi/4)$. Определите количество точек пересечения этих траекторий.

Ответ: 18.

Задание 9

Космический крейсер оказался под атакой космических пиратов и отправил сигнал SOS. Два отряда космодесантников, находящихся на базе, получили сигнал и направились на помощь.

От базы (узел А) до крейсера (узел Д) можно добраться через сеть червоточин, соединяющих некоторые узлы в пространстве (планеты или точки в пространстве).

Космическое пространство можно представить в виде графа с пятью вершинами, где вершины – это узлы (планеты или точки в пространстве), а рёбра – это червоточины с определённой длиной. Ниже в матрице смежности дана информация о длине рёбер-червоточин.

	А	Б	В	Г	Д
А	0	6	5	7	0
Б	6	0	4	3	9
В	5	4	0	1	8
Г	7	3	1	0	6
Д	0	9	8	6	0

Отряд Альфа может перемещаться только по рёбрам с чётной длиной.

Отряд Бета обязан начать движение с ребра с нечётной длиной и чередовать движение по рёбрам с чётной и нечётной длиной.

1) Какой из отрядов доберётся до крейсера быстрее? В случае равенства длин маршрутов выберите тот отряд, который прошёл через меньшее число узлов. В поле ответа введите

название отряда одним словом с большой буквы без дополнительных символов (Альфа или Бета). **Ответ:** Бета.

2) Возможно ли, что один из отрядов не сможет добраться до крейсера, учитывая ограничения на рёбра? Напишите в поле ответа знак «+», если это возможно, или знак «-» – в противном случае (без пробелов, кавычек или иных посторонних символов). **Ответ:** -.

3) Введите в поле ответа длину кратчайшего маршрута (одним числом без пробелов и дополнительных знаков). **Ответ:** 13.

Задание 10

Космическое агентство разработало программу на языке C++ для управления временем связи и запасами топлива на борту исследовательского космического корабля. Программа использует три функции, работающие с локальными и глобальными переменными, а также реализующие рекурсию.

Функция `calculateCommunicationTime` принимает расстояние до Земли в световых минутах и возвращает время, необходимое для передачи сигнала туда и обратно.

Функция `checkFuelLevel` принимает текущий уровень топлива и минимально допустимый уровень. Если уровень топлива опускается ниже 20% от начального запаса, выводится предупреждение.

Рекурсивная функция `calculateMaxTravelDistance` принимает текущий запас топлива, расход топлива на единицу расстояния и возвращает максимальное расстояние, которое корабль может преодолеть с оставшимся топливом.

Входные данные:

- Начальный запас топлива (переменная `fuelLevel`): 1000 тонн.
- Минимально допустимый уровень топлива (переменная `minFuelLevel`): 20% от начального запаса.
- Расход топлива на единицу расстояния (переменная `fuelConsumptionPerUnit`): 5 тонн.
- Расстояние до Земли (переменная `distanceToEarth`): 40 световых минут.
- Пройденное расстояние (переменная `travelDistance`): 100 единиц.

Вопросы:

1. Сколько времени потребуется для передачи сигнала до Земли и обратно? **Ответ:** 80.
2. Какой будет уровень топлива после полёта на 100 единиц расстояния? **Ответ:** 500.
3. Какое максимальное расстояние может преодолеть корабль с оставшимся топливом? **Ответ:** 100.

Код программы:

```
#include <iostream>
using namespace std;

int fuelLevel;

int calculateCommunicationTime(int distance) {
    int communicationTime = distance * 2;
    return communicationTime;
}

void checkFuelLevel(int fuel, int minFuelLevel) {
    if (fuel < minFuelLevel) {
        cout << "Внимание: Уровень топлива ниже допустимого!" << endl;
    }
}

int calculateMaxTravelDistance(int fuel, int fuelConsumptionPerUnit) {
    if (fuel < fuelConsumptionPerUnit) {
        return 0;
    }
    return 1 + calculateMaxTravelDistance(fuel - fuelConsumptionPerUnit,
    fuelConsumptionPerUnit);
}

int main() {
    fuelLevel = 1000;
    int minFuelLevel = 0.2 * fuelLevel;
    int fuelConsumptionPerUnit = 5;
    int distanceToEarth = 40;
    int travelDistance = 100;

    int communicationTime = calculateCommunicationTime(distanceToEarth);
    cout << "Время передачи сигнала до Земли и обратно: " << communicationTime << "
    минут" << endl;

    int fuelConsumed = travelDistance * fuelConsumptionPerUnit;
    fuelLevel -= fuelConsumed;
    checkFuelLevel(fuelLevel, minFuelLevel);
    cout << "Уровень топлива после полёта на " << travelDistance << " единиц
    расстояния: " << fuelLevel << " тонн" << endl;

    int maxDistance = calculateMaxTravelDistance(fuelLevel, fuelConsumptionPerUnit);
    cout << "Максимальное расстояние, которое корабль может ещё преодолеть: " <<
    maxDistance << " единиц" << endl;

    return 0;
}
```

Вариант 15

Задание 1

Жители планеты Линь для того, чтобы избежать сдвига привычных времён года, ввели понятие антивисокосного года, который на один день короче обычного. В обычном Линьском году ровно 290 Линьских дней. Антивисокосным является каждый 7-й год, а также каждый 8-й год. Если же очередной 7-й год совпадает с очередным 8-м, то год считается обычным. Найдите точную продолжительность года на планете Линь (выраженную в Линьских днях). Выберите из предложенных вариантов ответа, на сколько точная продолжительность года отличается от 290 дней.

Варианты ответа:

- А) 2/7
- Б) 1/8
- В) 7/145
- Г) 13/56

Ответ: Г) 13/56.

Задание 2

Космическое агентство разрабатывает автономного робота для исследования далёких планет. Робот должен уметь анализировать данные и выполнять повторяющиеся задачи в зависимости от полученных данных. Для этого разработана программа на языке C++, в которой используется несколько видов циклов и работа с массивами данных.

Входные данные:

- Робот получает данные о температуре на различных участках маршрута по поверхности планеты. Данные представлены в виде одномерного массива целых чисел (*temperatures[]*), где каждый элемент массива – это температура на конкретном участке.
- Робот фиксирует наличие препятствий на каждом участке, что представлено в виде другого массива (*obstacles[]*), где 1 означает наличие препятствия, а 0 – его отсутствие.

Роботу необходимо проанализировать данные и выполнить следующие задачи:

- 1) Робот должен двигаться по поверхности планеты, пока температура не поднимется выше 40°C или пока не встретит большое препятствие (обозначенное в массиве как 1). Определить, сколько участков робот пройдёт до остановки (переменная *count_before_stop*). Участок, на котором робот остановился, считать не нужно.
- 2) Робот должен проводить фотосъёмку каждого участка, где температура выше 0°C, и определить, сколько снимков сделает робот до остановки (переменная *photo_count*). На участке, где остановился робот, можно провести фотосъёмку.
- 3) Робот должен подсчитать общее количество участков с нормальной температурой (от -35°C до +35°C) и без препятствий (переменная *count_normal_temp*).

Просуммируйте три числа, получившихся в результате выполнения указанных задач, и выберите из предложенных ниже вариантов ответа, чему равна эта сумма:

- A) 6
- Б) 7
- В) 8**
- Г) 9

Код программы:

```
#include <iostream>
```

```
using namespace std;
```

```
int main() {
```

```
    int temperatures[] = {-25, -40, 40, 0, 45, 30, -20};
```

```
    int obstacles[] = {0, 0, 0, 0, 0, 1, 1};
```

```
    int n = sizeof(temperatures) / sizeof(temperatures[0]);
```

```
    int i = 0;
```

```
    while (i < n && temperatures[i] <= 40 && obstacles[i] == 0) {
```

```
        i++;
```

```
    }
```

```
    int count_before_stop = i;
```

```
    int j = 0;
```

```
    int photo_count = 0;
```

```
    do {
```

```
        if (temperatures[j] > 0) {
```

```
            photo_count++;
```

```
        }
```

```
        j++;
```

```
    } while (j <= count_before_stop);
```

```
    int count_normal_temp = 0;
```

```
    for (int k = 0; k < n; k++) {
```

```
        if (temperatures[k] >= -35 && temperatures[k] <= 35 && obstacles[k] == 0) {
```

```
            count_normal_temp++;
```

```
        }
```

```

}

cout << "Количество участков до остановки: " << count_before_stop << endl;

cout << "Количество снимков: " << photo_count << endl;

cout << "Количество участков с нормальной температурой и без препятствий: " <<
count_normal_temp << endl;

return 0;

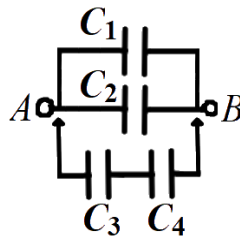
}

```

Ответ: В) 8.

Задание 3

При проведении серии опытов на борту космической станции батареею конденсаторов C_1 и C_2 от источника Э.Д.С. зарядили до разности потенциалов 15 В на клеммах **A-B** (см. рисунок).



Затем источник отключили и к клеммам **A-B** подключили батарею незаряженных конденсаторов C_3 и C_4 . Ёмкости конденсаторов: $C_1 = 0,10$ мкФ, $C_2 = 0,20$ мкФ, $C_4 = 0,05$ мкФ. Площадь обкладок плоского конденсатора C_3 равна $7,2 \cdot 10^{-2}$ м², расстояние между ними $1,68 \cdot 10^{-4}$ м, диэлектрическая проницаемость вещества, заполняющего пространство между обкладками, $\epsilon = 53$. Рассчитайте максимальную энергию конденсатора C_4 .

Приставка $1\text{мк} = 10^{-6}$; $\epsilon_0 = 8,85 \cdot 10^{-12}$ Ф/м.

Варианты ответа:

- А) 7,8 мкДж
- Б) 1,4 мкДж
- В) 5,6 мкДж
- Г) **2,8 мкДж**
- Д) 9,5 мкДж

Ответ: Г) 2,8 мкДж

Задание 4

На горизонтальном участке поверхности неизвестной планеты под углом 30° к горизонту толкнули ядро с начальной скоростью 16 м/с. Считая форму планеты шаром, её массу $3 \cdot 10^{20}$ кг, а величину первой космической скорости для этой планеты равной 458,9 м/с, определите дальность полета ядра. Результат представьте в виде числа «М·10² м» и выберите ответ с мантиссой «М».

Гравитационная постоянная $G = 6,67 \cdot 10^{-11} \text{ Н} \cdot \text{м}^2/\text{кг}^2$.

Возможные варианты ответа:

- А) 1
- Б) 2
- В) 3
- Г) 4
- Д) 5
- Е) 6
- Ж) 7

Ответ: А) 1

Задание 5

Некоторая планета совершает полный оборот вокруг своей звезды за 760 часов, а сутки (для наблюдателя на этой планете) длятся 38 часов. За сколько часов эта планета совершает один оборот вокруг своей оси, если направление вращения вокруг оси противоположно направлению вращения вокруг звезды? Выберите один вариант ответа из списка ниже с учётом округления.

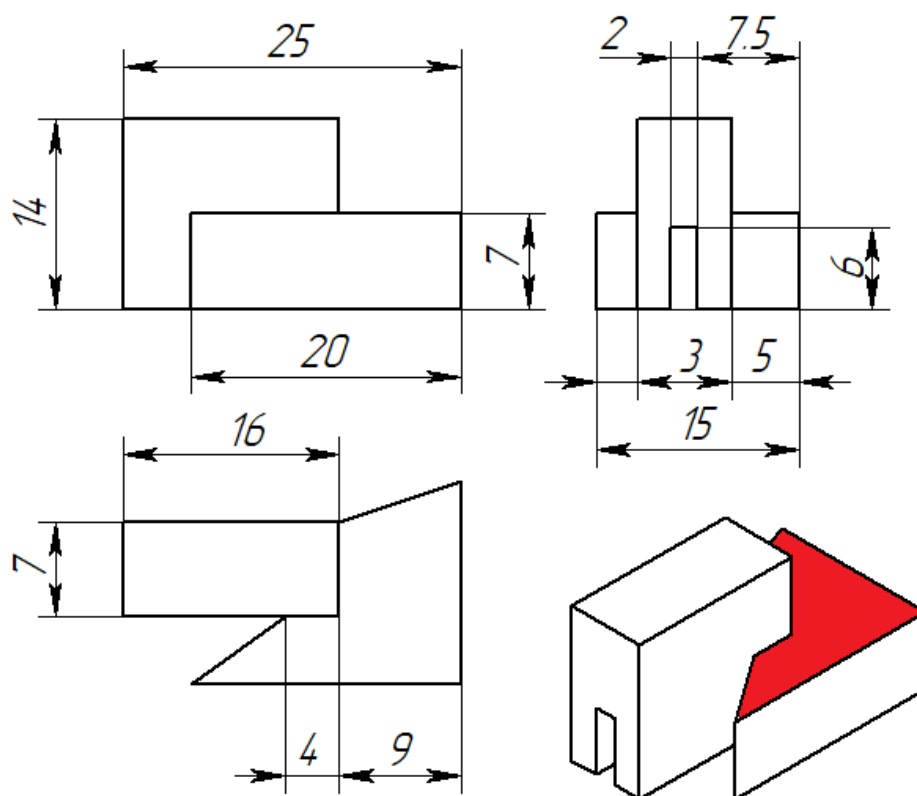
Варианты ответа:

- А) 42,22
- Б) 36,19
- В) 38,00
- Г) **40,00**

Ответ: Г) 40,00.

Задание 6

По размерам, имеющимся на чертеже, определите площадь выделенной цветом грани.



Варианты ответа:

- А) 153
- Б) 156
- В) 159**
- Г) 162

Ответ: В) 159

Задание 7

Найти первую и вторую космические скорости для планеты Венера. Масса Венеры составляет $4,9 \cdot 10^{24}$ кг, диаметр – 12 228 км. При расчётах принять гравитационную постоянную равной $6,67 \cdot 10^{-11}$ Н·м²/кг². Ответ выразите в км/с с одним знаком после запятой. При расчёте второй космической скорости необходимо воспользоваться полученным значением первой. В поля ответа необходимо ввести только числа.

Ответ: Первая космическая скорость: 7,3 км/с.

Вторая космическая скорость: 10,3 км/с.

Задание 8

Два тела движутся в одной плоскости. Если ввести на этой плоскости декартову систему координат, то траектории их движения совпадут с графиками функций $f(x) = \frac{(x-0.5)^2}{12} - 1$ и $g(x) = \cos(\pi x)$. Определите количество точек пересечения этих траекторий.

Ответ: 10.

Задание 9

Космический крейсер оказался под атакой космических пиратов и отправил сигнал SOS. Два отряда космодесантников, находящихся на базе, получили сигнал и направились на помощь.

От базы (узел А) до крейсера (узел Д) можно добраться через сеть червоточин, соединяющих некоторые узлы в пространстве (планеты или точки в пространстве).

Космическое пространство можно представить в виде графа с пятью вершинами, где вершины – это узлы (планеты или точки в пространстве), а рёбра – это червоточины с определённой длиной. Ниже в матрице смежности дана информация о длине рёбер-червоточин.

	А	Б	В	Г	Д
А	0	6	4	3	0
Б	6	0	2	9	5
В	4	2	0	1	8
Г	3	9	1	0	6
Д	0	5	8	6	0

Отряд Альфа может перемещаться только по рёбрам с чётной длиной.

Отряд Бета обязан начать движение с ребра с нечётной длиной и чередовать движение по рёбрам с чётной и нечётной длиной.

1) Какой из отрядов доберётся до крейсера быстрее? В случае равенства длин маршрутов выберите тот отряд, который прошёл через меньшее число узлов. В поле ответа введите название отряда одним словом с большой буквы без дополнительных символов (Альфа или Бета). **Ответ:** Бета.

2) Возможно ли, что один из отрядов не сможет добраться до крейсера, учитывая

ограничения на рёбра? Напишите в поле ответа знак «+», если это возможно, или знак «-» – в противном случае (без пробелов, кавычек или иных посторонних символов). **Ответ:** -.

3) Введите в поле ответа длину кратчайшего маршрута (одним числом без пробелов и дополнительных знаков). **Ответ:** 9.

Задание 10

Космическое агентство разработало программу на языке C++ для управления уровнем радиации и состоянием здоровья экипажа на борту исследовательского космического корабля. Программа использует три функции, работающие с локальными и глобальными переменными, а также реализующие рекурсию.

Функция `calculateRadiationExposure` принимает количество дней полёта и уровень радиации в космосе и возвращает общее количество радиации, которое получит экипаж за этот период.

Функция `checkRadiationLevel` принимает текущий уровень накопленной радиации и максимальный допустимый уровень. Если накопленная радиация превышает 70% от максимального допустимого уровня, выводится предупреждение.

Рекурсивная функция `calculateDaysUntilMaxRadiation` принимает текущий уровень накопленной радиации, ежедневный уровень радиации и возвращает количество дней, через которое будет достигнут максимальный допустимый уровень радиации.

Входные данные:

- Начальный уровень накопленной радиации (переменная `radiationLevel`): 0 единиц.
- Максимально допустимый уровень накопленной радиации (переменная `maxRadiationLevel`): 1000 единиц.
- Ежедневный уровень радиации в космосе (переменная `dailyRadiation`): 15 единиц.
- Количество дней полёта (переменная `flightDays`): 40 дней.

Вопросы:

1. Какое общее количество радиации получит экипаж за 40 дней полёта? **Ответ:** 600.
2. Какой будет уровень накопленной радиации после 40 дней полёта? **Ответ:** 600.
3. Через сколько дней будет достигнут максимальный допустимый уровень радиации? **Ответ:** 27.

Код программы:

```
#include <iostream>
using namespace std;
```

```
int radiationLevel;
```

```
int calculateRadiationExposure(int days, int dailyRadiation) {
    int totalRadiation = days * dailyRadiation;
```

```

    return totalRadiation;
}

void checkRadiationLevel(int radiation, int maxRadiationLevel) {
    if (radiation > 0.7 * maxRadiationLevel) {
        cout << "Внимание: Уровень радиации превышает 70% от максимального!" << endl;
    }
}

int calculateDaysUntilMaxRadiation(int currentRadiation, int dailyRadiation, int
maxRadiationLevel) {
    if (currentRadiation >= maxRadiationLevel) {
        return 0;
    }
    return 1 + calculateDaysUntilMaxRadiation(currentRadiation + dailyRadiation,
dailyRadiation, maxRadiationLevel);
}

int main() {
    radiationLevel = 0;
    int maxRadiationLevel = 1000;
    int dailyRadiation = 15;
    int flightDays = 40;

    int totalRadiation = calculateRadiationExposure(flightDays, dailyRadiation);
    cout << "Экипаж получит " << totalRadiation << " единиц радиации за " << flightDays
<< " дней полёта" << endl;

    radiationLevel += totalRadiation;
    checkRadiationLevel(radiationLevel, maxRadiationLevel);
    cout << "Уровень накопленной радиации после " << flightDays << " дней: " <<
radiationLevel << " единиц" << endl;

    int daysUntilMaxRadiation = calculateDaysUntilMaxRadiation(radiationLevel,
dailyRadiation, maxRadiationLevel);
    cout << "Максимальный допустимый уровень радиации будет достигнут через " <<
daysUntilMaxRadiation << " дней" << endl;
    return 0;
}

```

Вариант 16

Задание 1

Жители планеты Линь для того, чтобы избежать сдвига привычных времён года, ввели понятие антивисокосного года, который на один день короче обычного. В обычном Линьском году ровно 135 Линьских дней. Антивисокосным является каждый 3-й год, а также каждый 5-й год. Если же очередной 3-й год совпадает с очередным 5-м, то год считается обычным. Найдите точную продолжительность года на планете Линь (выраженную в Линьских днях). Выберите из предложенных вариантов ответа, на сколько точная продолжительность года отличается от 135 дней.

Варианты ответа:

- А) 2/3
- Б) 2/5**
- В) 8/15
- Г) 1/9

Ответ: Б) 2/5.

Задание 2

Космическое агентство разрабатывает автономного робота для исследования далёких планет. Робот должен уметь анализировать данные и выполнять повторяющиеся задачи в зависимости от полученных данных. Для этого разработана программа на языке C++, в которой используется несколько видов циклов и работа с массивами данных.

Входные данные:

- Робот получает данные о температуре на различных участках маршрута по поверхности планеты. Данные представлены в виде одномерного массива целых чисел (*temperatures[]*), где каждый элемент массива – это температура на конкретном участке.
- Робот фиксирует наличие препятствий на каждом участке, что представлено в виде другого массива (*obstacles[]*), где 1 означает наличие препятствия, а 0 – его отсутствие.

Роботу необходимо проанализировать данные и выполнить следующие задачи:

- 1) Робот должен двигаться по поверхности планеты, пока температура не поднимется выше 40°C или пока не встретит большое препятствие (обозначенное в массиве как 1). Определить, сколько участков робот пройдёт до остановки (переменная *count_before_stop*). Участок, на котором робот остановился, считать не нужно.
- 2) Робот должен проводить фотосъёмку каждого участка, где температура не выше 0°C, и определить, сколько снимков сделает робот до остановки (переменная *photo_count*). На участке, где остановился робот, можно провести фотосъёмку.
- 3) Робот должен подсчитать общее количество участков с нормальной температурой (от -35°C до +35°C) и без препятствий (переменная *count_normal_temp*).

Просуммируйте три числа, получившихся в результате выполнения указанных задач, и выберите из предложенных ниже вариантов ответа, чему равна эта сумма:

- A) 6
- Б) 7
- В) 8
- Г) 9

Код программы:

```
#include <iostream>
```

```
using namespace std;
```

```
int main() {
```

```
    int temperatures[] = {-25, -40, 40, 0, 45, 30, -20};
```

```
    int obstacles[] = {0, 0, 0, 0, 0, 1, 1};
```

```
    int n = sizeof(temperatures) / sizeof(temperatures[0]);
```

```
    int i = 0;
```

```
    while (i < n && temperatures[i] <= 40 && obstacles[i] == 0) {
```

```
        i++;
```

```
    }
```

```
    int count_before_stop = i;
```

```
    int j = 0;
```

```
    int photo_count = 0;
```

```
    do {
```

```
        if (temperatures[j] <= 0) {
```

```
            photo_count++;
```

```
        }
```

```
        j++;
```

```
    } while (j <= count_before_stop);
```

```
    int count_normal_temp = 0;
```

```
    for (int k = 0; k < n; k++) {
```

```
        if (temperatures[k] >= -35 && temperatures[k] <= 35 && obstacles[k] == 0) {
```

```
            count_normal_temp++;
```

```
        }
```

```

}

cout << "Количество участков до остановки: " << count_before_stop << endl;

cout << "Количество снимков: " << photo_count << endl;

cout << "Количество участков с нормальной температурой и без препятствий: " <<
count_normal_temp << endl;

return 0;

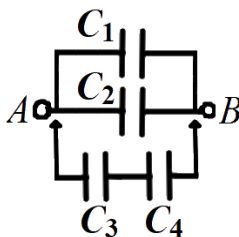
}

```

Ответ: Г) 9.

Задание 3

При проведении серии опытов на борту космической станции к клеммам **A-B** батареи конденсаторов C_1 и C_2 подключили источник Э.Д.С. (см. рисунок).



Когда энергия конденсатора C_2 достигла $1,25 \text{ мкДж}$, источник отключили и к клеммам **A-B** подключили батарею незаряженных конденсаторов C_3 и C_4 . Ёмкости конденсаторов $C_1 = 0,023 \text{ мкФ}$, $C_2 = 0,013 \text{ мкФ}$, $C_3 = 0,080 \text{ мкФ}$. Площадь обкладок плоского конденсатора C_4 равна $6,38 \cdot 10^{-3} \text{ м}^2$, расстояние между ними $1,27 \cdot 10^{-5} \text{ м}$, диэлектрическая проницаемость вещества, заполняющего пространство между обкладками, $\epsilon = 72$. Рассчитайте разность потенциалов на клеммах **A-B** при максимальной зарядке конденсаторов блока C_{34} .

Приставка $1 \text{ мк} = 10^{-6}$; $\epsilon_0 = 8,85 \cdot 10^{-12} \text{ Ф/м}$.

Варианты ответа:

- А) 7,0 В
- Б) 3,1 В
- В) 5,0 В**
- Г) 9,9 В
- Д) 6,2 В

Ответ: В) 5,0 В.

Задание 4

С горизонтальной площадки на поверхности неизвестной планеты под углом 60° толкнули ядро с начальной скоростью $15,0 \text{ м/с}$. Через несколько секунд оно упало на расстоянии 75

м от начального положения. Учитывая результаты данного эксперимента, считая форму планеты шаром, а первую космическую скорость для неё равной 193 м/с, рассчитайте массу планеты. Результат представьте в виде «М·10¹⁸ кг» и запишите в ответ мантиссу «М».

Гравитационная постоянная $G = 6,67 \cdot 10^{-11} \text{ Н} \cdot \text{м}^2/\text{кг}^2$.

Возможные варианты ответа:

- А) 3
- Б) 4
- В) 5
- Г) 6
- Д) 7
- Е) 8**
- Ж) 9

Ответ: Е) 8

Задание 5

Некоторая планета совершает полный оборот вокруг своей звезды за 600 часов, а сутки (для наблюдателя на этой планете) длятся 30 часов. За сколько часов эта планета совершает один оборот вокруг своей оси, если направление вращения вокруг оси противоположно направлению вращения вокруг звезды? Выберите один вариант ответа из списка ниже с учётом округления.

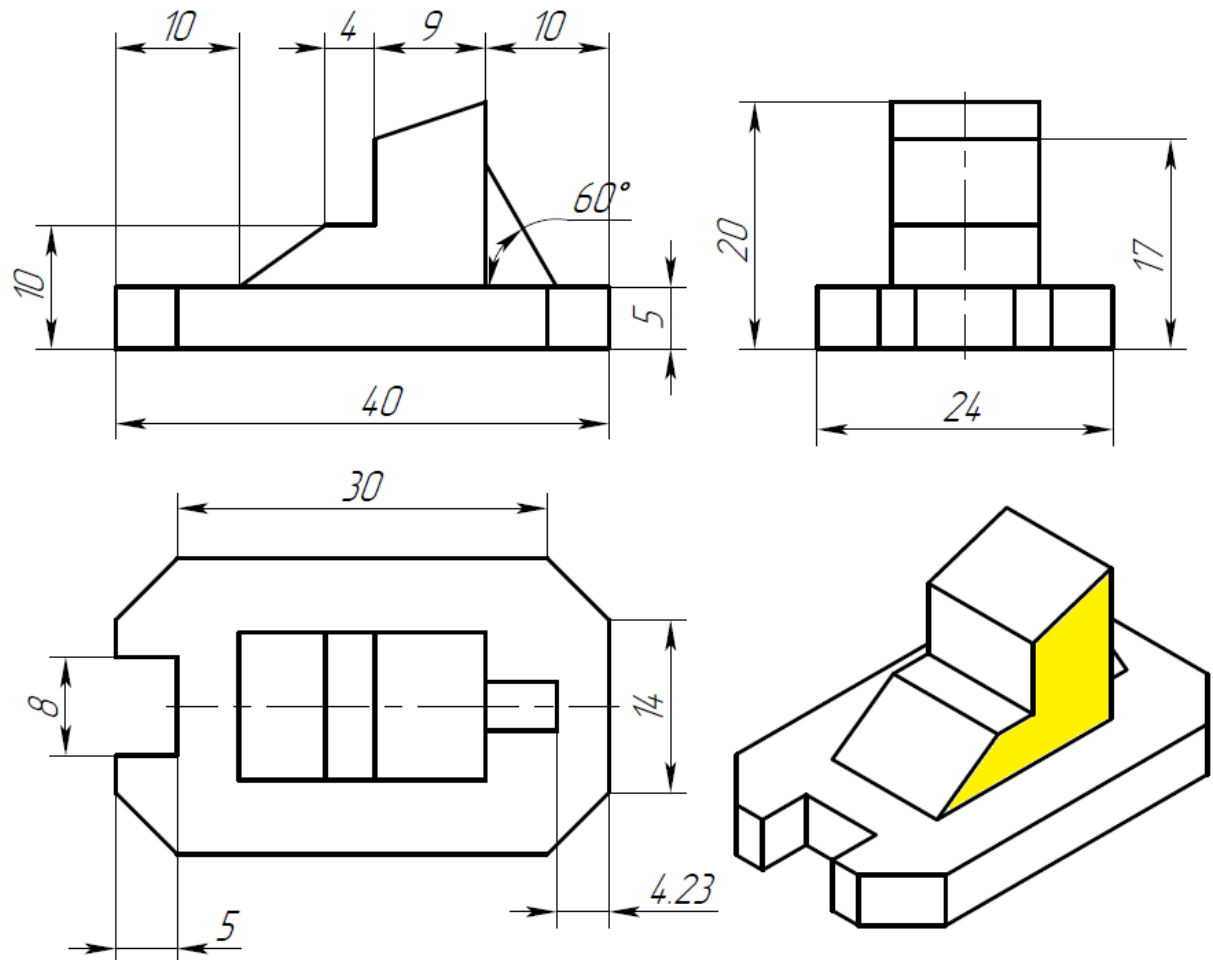
Варианты ответа:

- А) 31,58**
- Б) 33,33
- В) 30,00
- Г) 28,57

Ответ: А) 31,58.

Задание 6

По размерам, имеющимся на чертеже, определите площадь выделенной цветом грани.



Варианты ответа:

- А) 153
- Б) 156
- В) 159**
- Г) 162

Ответ: В) 159

Задание 7

Найти первую и вторую космические скорости для планеты Нептун. Масса Нептуна составляет $1,0 \cdot 10^{26}$ кг, диаметр – 44 600 км. При расчётах принять гравитационную постоянную равной $6,67 \cdot 10^{-11}$ Н·м²/кг². Ответ выразите в км/с с одним знаком после запятой. При расчёте второй космической скорости необходимо воспользоваться полученным значением первой. В поля ответа необходимо ввести только числа.

Ответ: Первая космическая скорость: 17,3 км/с.

Вторая космическая скорость: 24,5 км/с.

Задание 8

Два тела движутся в одной плоскости. Если ввести на этой плоскости декартову систему координат, то траектории их движения совпадут с графиками функций $f(x) = \frac{(x-2)^2}{6} - 1$ и $g(x) = \cos(2\pi x)$. Определите количество точек пересечения этих траекторий.

Ответ: 14.

Задание 9

Космический крейсер оказался под атакой космических пиратов и отправил сигнал SOS. Два отряда космодесантников, находящихся на базе, получили сигнал и направились на помощь.

От базы (узел А) до крейсера (узел Д) можно добраться через сеть червоточин, соединяющих некоторые узлы в пространстве (планеты или точки в пространстве).

Космическое пространство можно представить в виде графа с пятью вершинами, где вершины – это узлы (планеты или точки в пространстве), а рёбра – это червоточины с определённой длиной. Ниже в матрице смежности дана информация о длине рёбер-червоточин.

	А	Б	В	Г	Д
А	0	2	7	5	0
Б	2	0	4	3	9
В	7	4	0	1	5
Г	5	3	1	0	5
Д	0	9	5	5	0

Отряд Альфа может перемещаться только по рёбрам с чётной длиной.

Отряд Бета обязан начать движение с ребра с нечётной длиной и чередовать движение по рёбрам с чётной и нечётной длиной.

1) Какой из отрядов доберётся до крейсера быстрее? В случае равенства длин маршрутов выберите тот отряд, который прошёл через меньшее число узлов. В поле ответа введите название отряда одним словом с большой буквы без дополнительных символов (Альфа или Бета). **Ответ:** Бета.

2) Возможно ли, что один из отрядов не сможет добраться до крейсера, учитывая

ограничения на рёбра? Напишите в поле ответа знак «+», если это возможно, или знак «-» – в противном случае (без пробелов, кавычек или иных посторонних символов). **Ответ:** +.

3) Введите в поле ответа длину кратчайшего маршрута (одним числом без пробелов и дополнительных знаков). **Ответ:** 20.

Задание 10

Космическое агентство разработало программу на языке C++ для управления запасами энергии и анализа работы двигателей на борту исследовательского космического корабля. Программа использует три функции, работающие с локальными и глобальными переменными, а также реализующие рекурсию.

Функция `calculateEngineUsage` принимает количество двигателей, количество дней полёта и ежедневное потребление энергии одним двигателем, возвращает общее количество энергии, которое потребят двигатели за этот период.

Функция `checkEnergyLevel` принимает текущий уровень энергии и минимально допустимый уровень. Если уровень энергии опускается ниже 15% от начального запаса, выводится предупреждение.

Рекурсивная функция `calculateDaysUntilEnergyDepletion` принимает текущий уровень энергии и ежедневное общее потребление энергии, возвращая количество дней, на которые хватит оставшегося запаса энергии.

Входные данные:

- Начальный уровень энергии (переменная `energyLevel`): 8000 единиц.
- Минимально допустимый уровень энергии (переменная `minEnergyLevel`): 15% от начального запаса.
- Ежедневное потребление энергии одним двигателем (переменная `dailyEngineConsumption`): 200 единиц.
- Количество двигателей (переменная `engines`): 3 двигателя.
- Количество дней полёта (переменная `flightDays`): 12 дней.

Вопросы:

1. Сколько энергии потребят двигатели за 12 дней полёта? **Ответ:** 7200.
2. Какой будет уровень энергии после 12 дней полёта? **Ответ:** 800.
3. На сколько дней хватит оставшегося запаса энергии? **Ответ:** 1.

Код программы:

```
#include <iostream>
using namespace std;

int energyLevel;
```

```

int calculateEngineUsage(int engines, int days, int dailyConsumption) {
    int energyUsed = engines * days * dailyConsumption;
    return energyUsed;
}

void checkEnergyLevel(int energy, int minEnergyLevel) {
    if (energy < minEnergyLevel) {
        cout << "Внимание: Уровень энергии ниже допустимого!" << endl;
    }
}

int calculateDaysUntilEnergyDepletion(int energy, int dailyConsumption) {
    if (energy < dailyConsumption) {
        return 0;
    }
    return 1 + calculateDaysUntilEnergyDepletion(energy - dailyConsumption,
dailyConsumption);
}

int main() {
    energyLevel = 8000;
    int engines = 3;
    int dailyEngineConsumption = 200;
    int flightDays = 12;
    int minEnergyLevel = 0.1 * energyLevel;

    int energyUsed = calculateEngineUsage(engines, flightDays, dailyEngineConsumption);
    cout << "Двигатели потребуют " << energyUsed << " единиц энергии за " << flightDays
<< " дней полёта" << endl;

    energyLevel -= energyUsed;
    checkEnergyLevel(energyLevel, minEnergyLevel);
    cout << "Уровень энергии после " << flightDays << " дней: " << energyLevel << "
единиц" << endl;

    int totalDailyConsumption = engines * dailyEngineConsumption;
    int daysUntilDepletion = calculateDaysUntilEnergyDepletion(energyLevel,
totalDailyConsumption);
    cout << "Оставшегося запаса энергии хватит на " << daysUntilDepletion << " дней"
<< endl;

    return 0;
}

```