

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ  
ОЛИМПИАДА ШКОЛЬНИКОВ**  
**Заключительный этап**  
**Продуктовый сектор**  
**Междисциплинарные задачи**  
**8 класс**

---

**Задача: Экспедиция к Марсу**

Ограничение времени 1 секунда

Ограничение памяти 64.0 Мб

Ввод стандартный ввод или input.txt

Вывод стандартный вывод или output.txt

Вы руководите миссией по отправке космического корабля к Марсу. Корабль состоит из  $N$  модулей ( $1 \leq N \leq 100$ ), каждый из которых имеет определенный вес и научную ценность. Ваша задача - выбрать подмножество этих модулей так, чтобы суммарный вес не превышал грузоподъемность корабля, а научная ценность была максимальной. Каждый модуль можно взять только один раз.

**Формат ввода**

Первая строка содержит два целых числа  $N$  и  $W$ , где  $N$  - количество модулей, а  $W$  - грузоподъемность корабля. Следующие  $N$  строк содержат по два целых числа: вес и научную ценность каждого модуля.

**Формат вывода**

Программа должна вывести одно целое число - максимальную научную ценность модулей, которые можно взять на борт, не превышая грузоподъемности корабля. Если модулей, подходящих по весу, не найдено - вывести 0.

**Пример**

**Ввод**    **Вывод**

```
5 20
5 4
7 6    21
3 5
12 12
6 8
```

**Примечания**

В примере оптимальным выбором модулей будет взять модули с весом 5 (ценность 4), 3 (ценность 5) и 12 (ценность 12), что дает суммарный вес 20 и научную ценность 21.

**Пример решения**

```
def max_scientific_value(N, W, modules):
    dp = [[0 for _ in range(W + 1)] for _ in range(N + 1)]
    for i in range(1, N + 1):
        for w in range(1, W + 1):
            weight, value = modules[i - 1]
            if weight > w:
                dp[i][w] = dp[i - 1][w]
            else:
```

МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ  
ОЛИМПИАДА ШКОЛЬНИКОВ  
Заключительный этап  
Продуктовый сектор  
Междисциплинарные задачи  
8 класс

---

```
dp[i][w] = max(dp[i - 1][w], dp[i - 1][w - weight] + value)
return dp[N][W]
```

```
N, W = map(int, input().split())
modules = []
for _ in range(N):
    weight, value = map(int, input().split())
    modules.append((weight, value))

print(max_scientific_value(N, W, modules))
```

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ  
ОЛИМПИАДА ШКОЛЬНИКОВ**  
**Заключительный этап**  
**Продуктовый сектор**  
**Междисциплинарные задачи**  
**8 класс**

---

**Задача: Мобильный трехколесный робот**

Ограничение времени 1 секунда

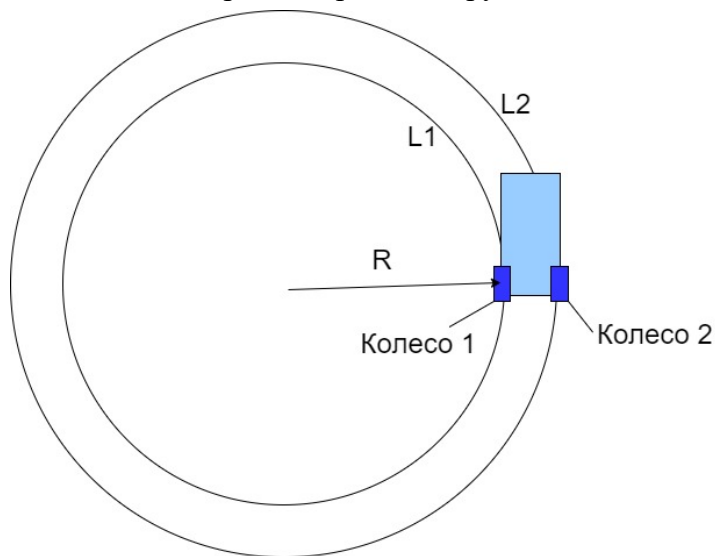
Ограничение памяти 64.0 Мб

Ввод стандартный ввод или input.txt

Вывод стандартный вывод или output.txt

Перед роботом стоит задача вычисления скорости перемещения ведущих колес робота за заданное время  $t$  при равномерном перемещении робота по окружности.

Робот движется равномерно по окружности как показано на рисунке ниже.



**Формат ввода**

На вход программы поступает 3 строки, содержащие вещественные числа:

Строка 1 содержит значение расстояния от центра окружности до левого колеса робота  $R$  (измеряется в сантиметрах).

Строка 2 содержит значение межосевого расстояния между колесами робота  $k$  (измеряется в сантиметрах).

Строка 3 содержит значение времени перемещения робота  $t$  (измеряется в секундах).

**Формат вывода**

На выходе программное обеспечение должно выдавать рассчитанные значения скорости ведущих колес робота.

Значение скорости первого колеса робота записывается в первую строку. Значение указывается в см/с. Значение необходимо округлить до двух знаков после запятой.

Значение скорости колеса робота записывается во вторую строку. Значение указывается в см/с. Значение необходимо округлить до двух знаков после запятой.

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ  
ОЛИМПИАДА ШКОЛЬНИКОВ**  
**Заключительный этап**  
**Продуктовый сектор**  
**Междисциплинарные задачи**  
**8 класс**

---

**Пример 1**

<b>Ввод</b>	<b>Вывод</b>
10.00	
4.00	10.47
6.00	14.66

**Пример 2**

<b>Ввод</b>	<b>Вывод</b>
11.09	
9.69	1.74
40.00	3.26

**Пример 3**

<b>Ввод</b>	<b>Вывод</b>
10.51	
5.63	1.57
42.00	2.41

**Пример решения**

```
import math
```

```
class Robot:
```

```
    def __init__(self, circ_radius, wheelbase, time_сек):  
        self.circ_radius = circ_radius # радиус круга  
        self.wheelbase = wheelbase # межосевое расстояние  
        self.time_сек = time_сек # время движения
```

```
    def calculate_circumferences(self):
```

```
        R = self.circ_radius  
        k = self.wheelbase  
        t = self.time_сек  
        L_1 = 2 * math.pi * R  
        L_2 = 2 * math.pi * (R + k)
```

```
    return L_1, L_2
```

```
    def calculate_speeds(self, L_1, L_2):
```

```
        t = self.time_сек  
        V_1 = L_1 / t  
        V_2 = L_2 / t
```

```
    return V_1, V_2
```

```
with open("input.txt", "r") as file:
```

```
    lines = file.readlines()
```

```
    circ_radius = float(lines[0].strip()) # радиус круга в сантиметрах
```

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ  
ОЛИМПИАДА ШКОЛЬНИКОВ**  
**Заключительный этап**  
**Продуктовый сектор**  
**Междисциплинарные задачи**  
**8 класс**

---

```
wheelbase = float(lines[1].strip()) # межосевое расстояние в сантиметрах  
time_сек = float(lines[2].strip()) # время движения робота в секундах
```

```
my_robot = Robot(circ_radius, wheelbase, time_сек)
```

```
L_1, L_2 = my_robot.calculate_circumferences()
```

```
V_1, V_2 = my_robot.calculate_speeds(L_1, L_2)
```

```
with open("output.txt", "w") as file:
```

```
    file.write(str(round(V_1, 2)) + "\n" + str(round(V_2, 2)))
```

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ  
ОЛИМПИАДА ШКОЛЬНИКОВ**  
**Заключительный этап**  
**Продуктовый сектор**  
**Междисциплинарные задачи**  
**8 класс**

---

**Задача: Новая неизвестная планета**

Ограничение времени 1 секунда

Ограничение памяти 64.0 Мб

Ввод стандартный ввод или input.txt

Вывод стандартный вывод или output.txt

Астронавты высадились на новую неизвестную планету, поверхность которой покрыта жидкостью плотностью  $\rho_0$  м<sup>3</sup>. Со дна (глубина  $h$  м) щупом подняли образец грунта объёмом  $V$  м<sup>3</sup> и плотностью  $\rho_1$  м<sup>3</sup>. Работа по подъёму образца грунта на поверхность равна  $A$  Дж.

Определить ускорение свободного падения  $g$  (м/с<sup>2</sup>), действующее на поверхности этой неизвестной планеты.

**Формат ввода**

На вход подаётся пять вещественных чисел,  $\rho_0$ ,  $\rho_1$ ,  $V$ ,  $h$ ,  $A$ , каждое число подаётся с новой строки.

**Формат вывода**

Вывести ускорение свободного падения  $g$ , действующее на поверхности этой неизвестной планеты. Результат округлить до сотых.

**Пример**

**Ввод**    **Вывод**

```
995.7
1616.1
0.9     17.01
2
19000
```

**Пример решения**

```
rho0 = float(input())
rho1 = float(input())
V = float(input())
h = float(input())
A = float(input())
g = A / (h * V * (rho1 - rho0))
print(round(g, 2))
```

**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ  
ОЛИМПИАДА ШКОЛЬНИКОВ**  
**Заключительный этап**  
**Продуктовый сектор**  
**Междисциплинарные задачи**  
**8 класс**

---

**Задача: Калориметр**

Ограничение времени 1 секунда

Ограничение памяти 64.0 Мб

Ввод стандартный ввод или input.txt

Вывод стандартный вывод или output.txt

В калориметре находится смесь  $m_1$  кг льда и  $m_2$  кг воды в состоянии теплового равновесия.

В калориметр доливают  $m_3$  кг воды при температуре  $t$  (в градусах цельсия).

Найдите установившуюся температуру (в градусах цельсия) смеси после добавления воды, округленную до двух знаков после запятой.

Ответ округлить до 2 знаков после запятой.

Удельная теплота плавления льда – 330 кДж/кг. Удельная теплоёмкость воды – 4200 Дж/кг\*С.

**Формат ввода**

На вход подается четыре вещественных числа, разделенных пробелами:  $m_1$ ,  $m_2$ ,  $m_3$ ,  $t$ .

**Формат вывода**

Необходимо вывести установившуюся температуру (в цельсиях) смеси после добавления воды.

Ответ округлить до 2 знаков после запятой.

**Пример**

Ввод	Вывод
0.1 0.1 5 100	94.64

**Примечания**

Если выходным значением является целое число, его необходимо выводить с дробной частью, равной нулю и двумя знаками в дробной части.

Разделителем является точка.

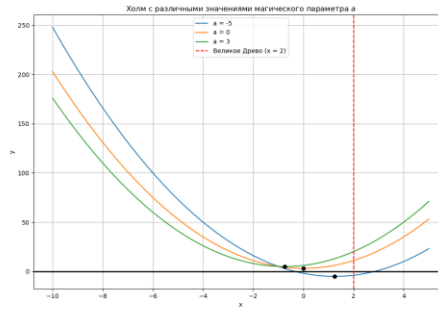
В случае получения значений: 1.0, 1, 1.000, 1.00000000000001, выводить следует: 1.00.

**Пример решения**

```
m1, m2, m3, t = [float(x) for x in input().split()]
if m1 * 330000 > m3 * 4200 * t:
    print('0.00')
else:
    a = 4200 * m3 * t - 330000 * m1
    b = 4200 * (m1 + m2 + m3)
    print(round(a / b, 2))
```

МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ  
ОЛИМПИАДА ШКОЛЬНИКОВ  
Заключительный этап  
Продуктовый сектор  
Междисциплинарные задачи  
8 класс

---





**МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ  
ОЛИМПИАДА ШКОЛЬНИКОВ**  
**Заключительный этап**  
**Продуктовый сектор**  
**Междисциплинарные задачи**  
**8 класс**

---

**Задача: Тайна Холма**

Ограничение времени 1 секунда

Ограничение памяти 64.0 Мб

Ввод стандартный ввод или input.txt

Вывод стандартный вывод или output.txt

В далекой волшебной стране, где геометрия и магия неразделимы, на землях могущественных Матемагов возвышаются волшебные холмы, формы которых описываются загадочными уравнениями. Самым загадочным из них является Холм, форма которого изменяется в зависимости от магического параметра  $a$ . Великий Матемаг дал задание своим ученикам: найти такие значения магического параметра  $a$ , при которых вершина Холма будет расположена на западе от Великого Древа (чья абсцисса равна двум) и при этом расстояние между двумя точками пересечения с Землей будет не менее длины квадратного корня из шести. Только найдя эти значения, ученики смогут открыть секрет Вечной Мудрости. Уравнение формы Холма задается как  $y = 2x^2 + ax + a + 3$ .

**Формат ввода**

Ввод состоит из одного числа - значения магического параметра  $a$ .

**Формат вывода**

- Вывести "First interval", если  $a$  попадает в левый допустимый интервал.
- Вывести "Second interval", если  $a$  попадает в правый интервал.
- Вывести "Not included in the intervals", если  $a$  не удовлетворяет ни одному из интервалов.

**Пример**

Ввод	Вывод
-5	First interval

**Примечания**

**Пример решения**

```
def check_conditions(a):  
    if a < -4 and a > -8:  
        return "First interval"  
    elif a > 12:  
        return "Second interval"  
    else:  
        return "Not included in the intervals"  
  
def main():  
    a = int(input())  
    answer = check_conditions(a)  
    print(answer)
```

МОСКОВСКАЯ ПРЕДПРОФЕССИОНАЛЬНАЯ  
ОЛИМПИАДА ШКОЛЬНИКОВ  
Заключительный этап  
Продуктовый сектор  
Междисциплинарные задачи  
8 класс

---

```
if __name__ == "__main__":  
    main()
```