

Всероссийская олимпиада школьников по информатике, 2016/17 уч. год
Первый (школьный) этап, г. Москва
Разбор заданий для 7–8 классов

Каждая задача оценивается в 10 баллов. Итоговый балл выставляется как сумма баллов за 4 задачи с лучшим результатом (то есть для получения максимального балла нужно решить 4 любые задачи).

Задача 1. Из Петербурга в Москву

Из Москвы в Санкт-Петербург выезжает пассажирский поезд, который движется со скоростью v км/ч. Одновременно из Санкт-Петербурга в Москву отправляется скоростной поезд «Сапсан», который движется со скоростью w км/ч. Расстояние от Москвы до Санкт-Петербурга равно s км. Определите, какое расстояние проедет «Сапсан» до встречи с пассажирским поездом. Считайте, что скорости движения поездов постоянные, временем остановок следует пренебречь.

Ответом на эту задачу является некоторое выражение, которое может содержать целые числа, переменные v , w и s (записываемые английскими буквами), операции сложения (обозначается «+»), вычитания (обозначается «-»), умножения (обозначается «*»), деления (обозначается «/») и круглые скобки для изменения порядка действий. Запись вида « $2s$ » для обозначения произведения числа 2 и переменной s неверная, нужно писать « $2 * s$ ».

Пример правильного (по форме записи) выражения: $v + (s - w) * 2$.

Решение

Поезда сближаются со скоростью $v + w$ и встретятся через время $s / (v + w)$. «Сапсан» за это время пройдёт расстояние $w * s / (v + w)$.

Ответ: $w * s / (v + w)$.

Задача 2. Упорядочите список

Дан список чисел:

3, 7, 1, 6, 2, 4, 8, 5.

Разрешается за одну операцию поменять местами два любых числа в этом списке. Например, если поменять местами числа 6 и 8, то получится список 3, 7, 1, 8, 2, 4, 6, 5.

Упорядочите этот список по возрастанию, то есть получите из него список 1, 2, 3, 4, 5, 6, 7, 8, используя минимальное число обменов.

Решение этой задачи нужно записать в виде последовательности обменов, каждый обмен записывается в одной строке. Один обмен записывается в виде двух различных чисел от 1 до 8, которые нужно поменять местами, записанных через пробел (пример: 6 8). Чем меньше обменов будет содержать ваше решение, тем больше баллов вы получите (при условии, что предложенный порядок обмена действительно упорядочивает список).

Решение

Наилучшее решение содержит пять обменов:

1 3
2 7
4 6
5 7
7 8

Задача 3. Ремонт моста

Мост через реку стоит на 15 опорах, обозначим их буквами латинского алфавита от А до О. Необходимо произвести ремонт опор моста, при разработке проекта ремонта была определена стоимость ремонта каждой опоры.

| Опора | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O |
|-------------------|----|---|----|----|----|----|----|----|----|----|-----|---|----|---|----|
| Стоимость ремонта | 10 | 5 | 20 | 10 | 30 | 40 | 30 | 30 | 30 | 50 | 100 | 2 | 10 | 3 | 20 |

Для того чтобы мост был надёжным, можно отремонтировать только часть опор, но с соблюдением следующих условий.

1. Крайние опоры (А и О) должны быть отремонтированы.
2. Не должно остаться двух стоящих рядом неотремонтированных опор.

Составьте план ремонта моста, при котором мост будет надёжным, то есть будут выполнены перечисленные выше условия, а стоимость ремонта будет минимальной.

Решение этой задачи нужно записать в виде перечисления букв, соответствующих опорам моста, которые нужно отремонтировать. Буквы нужно записывать в алфавитном порядке, каждая буква в ответе может встречаться не более одного раза. Например, ответ ABCDEFGHIJKLMNOP будет обозначать, что все опоры моста будут отремонтированы, но этот ответ, очевидно, не будет наилучшим. Чем меньше будет стоимость предложенного вами ремонта, тем больше баллов вы получите (при условии, что предложенный план ремонта является надёжным).

Решение

Наилучшее решение: ABDFHJLNO. Стоимость такого решения 170.

Задача 4. Строительство

Три мальчика – Боря, Коля и Слава – хотят помочь малышам построить замок из кубиков трёх цветов: белого, красного и синего. Замок имеет три башни, каждая башня будет состоять из пяти кубиков и должна иметь следующий вид (Б – белый кубик, К – красный, С – синий).

| | | |
|---|---|---|
| С | К | К |
| Б | С | К |
| Б | С | К |
| Б | Б | С |
| К | С | Б |
| 1 | 2 | 3 |

За одну секунду каждый мальчик может поставить кубик наверх одной из башни (в самом начале башни не содержат кубиков). При этом Боря использует кубики белого цвета, Коля – красного цвета, а Слава – синего. Два мальчика не могут одновременно ставить кубики в одну и ту же башню.

Составьте алгоритм действий мальчиков, при котором они построят замок за наименьшее время.

Решение этой задачи нужно записать в виде строки, каждая строка содержит описание действий мальчиков в очередную секунду. Первый символ является описанием действия Бори, второй символ – действие Коли, третий символ – действие Славы. Действие может быть одним из трёх чисел «1», «2», «3», что означает, что мальчик ставит свой кубик в башню

с соответствующим номером, если же мальчик в эту секунду ничего не делает, то вместо числа нужно написать символ «-». Например, строка «3-2» означает, что Боря ставит белый кубик в башню 3, Коля ничего не делает, а Слава ставит синий кубик в башню 2.

Чем короче будет ваш алгоритм, тем больше баллов вы получите (если, конечно же, алгоритм будет верным).

Решение

Наилучшее решение возможно за 6 секунд.

312
2-3
132
132
12-
-31

Решением задач 5–7 является программа, написанная на одном из языков программирования.

Задачи 5–7 необязательно решать для получения полного балла.

Ограничение по времени работы программы в задачах 5–7: 1 секунда.

Задания принимаются на проверку и оцениваются, только если они выдают правильный ответ на всех примерах входных и выходных данных, приведённых в условии задачи. Программа не должна выводить никаких иных сообщений, кроме того, что требуется найти в задаче.

Задача 5. Покупка

Ручка стоила K рублей. Первого сентября стоимость ручки увеличилась ровно на P процентов. Определите, сколько ручек можно купить на S рублей после подорожания.

Программа получает на вход три целых положительных числа. Первое число K – стоимость ручки в рублях до подорожания. Второе число P – величина подорожания ручки в процентах. Третье число S – имеющаяся сумма денег. Числа K и S не превосходят 10^7 , число P не превосходит 100.

Пример входных и выходных данных

| Ввод | Вывод | Примечание |
|----------------|-------|--|
| 33 5 100 | 2 | Ручка стоила 33 рубля. После подорожания на 5 % ручка будет стоить 34 рубля 65 копеек (заметим, что, поскольку первоначальная цена ручки была целым числом рублей, после подорожания стоимость ручки будет выражаться целым числом рублей и копеек). На 100 рублей после подорожания можно купить 2 ручки. |

Система оценивания

Решение, правильно работающее только для случаев, когда числа K и S не превосходят 100, будет оцениваться в 6 баллов.

Решение

Посчитаем стоимость ручки в копейках после подорожания — она будет равна $K * 100 + K * P$ (один процент стоимости ручки составляет K копеек). Затем переведем S рублей в копейки (умножив на 100) и поделим нацело на стоимость ручки после подорожания.

Пример решения на языке Python

```
k = int(input())
p = int(input())
s = int(input())
k = 100 * k + p * k
print(100 * s // k)
```

Типичной ошибкой в этой задаче было использование действительных чисел для представления стоимости ручки после подорожания (если стоимость ручки хранить в рублях). Поскольку действительные числа представляются неточно, то после деления значения S на действительное число $K + K * P / 100$ может получиться не целое число, а число, которое чуть меньше правильного целого результата, и в результате преобразования частного к целому числу путём отбрасывания дробной части ответ получался на 1 меньше правильного. Такие решения набирали, как правило, 7 баллов.

Пример решения с использованием действительных чисел (7 баллов)

```
k = float(input())
p = int(input())
s = int(input())
k = k + k * p / 100
print(int(s / k))
```

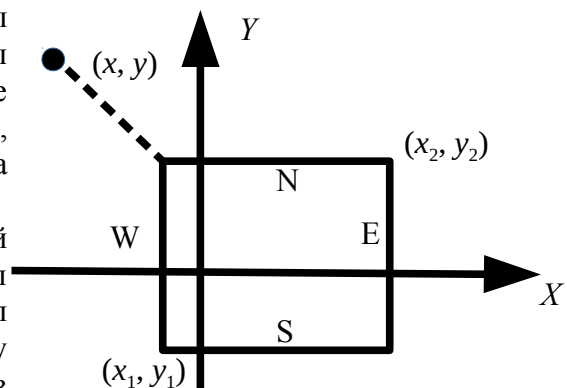
Задача 6. Плот

Посередине озера плавает плот, имеющий форму прямоугольника. Стороны плота направлены вдоль параллелей и меридианов. Введём систему координат, в которой ось OX направлена на восток, а ось OY — на север. Пусть юго-западный угол плота имеет координаты (x_1, y_1) , северо-восточный угол — координаты (x_2, y_2) .

Пловец находится в точке с координатами (x, y) . Определите, к какой стороне плота (северной, южной, западной или восточной) или к какому углу плота (северо-западному, северо-восточному, юго-западному, юго-восточному) пловцу нужно плыть, чтобы как можно скорее добраться до плота.

Программа получает на вход шесть чисел в следующем порядке: x_1, y_1 (координаты юго-западного угла плота), x_2, y_2 (координаты северо-восточного угла плота), x, y (координаты пловца). Все числа целые и по модулю не превосходят 100. Гарантируется, что $x_1 < x_2, y_1 < y_2, x \neq x_1, x \neq x_2, y \neq y_1, y \neq y_2$, координаты пловца находятся вне плота.

Если пловцу следует плыть к северной стороне плота, программа должна вывести символ «N», к южной — символ «S», к западной — символ «W», к восточной — символ «E». Если пловцу следует плыть к углу плота, нужно вывести одну из следующих строк: «NW», «NE», «SW», «SE».



Пример входных и выходных данных

| Ввод | Вывод | Примечание |
|-------------------------------|-------|--|
| -1 -2 5 3 -4 6 | NW | Картинка выше соответствует этому примеру. |

Система оценивания

Решение, правильно работающее для случаев, когда ответом является одна из сторон плота «N», «S», «W», «E», будет оцениваться в 6 баллов.

Решение, правильно работающее для случаев, когда ответом является один из углов «NW», «NE», «SW», «SE», будет оцениваться в 4 балла.

Решение

В этой задаче требовалось перебрать все возможные случаи расположения пловца относительно плота. Отметим на рисунке, что должна вывести программа для всех областей:

| | | |
|----|------|----|
| NW | N | NE |
| W | плот | E |
| SW | S | SE |

Например, программа должна вывести NE при условиях $x > x_2$ и $y > y_2$, вывести N при условиях $y > y_2$ и $x_1 < x < x_2$ и т. д. Необходимо аккуратно разобрать все случаи.

Пример решения на языке Python

```
x1 = int(input())
y1 = int(input())
x2 = int(input())
y2 = int(input())
x = int(input())
y = int(input())
if x > x2 and y > y2:
    print("NE")
elif x > x2 and y < y1:
    print("SE")
elif x < x1 and y < y1:
    print("SW")
elif x < x1 and y > y2:
    print("NW")
elif y > y2:
    print("N")
elif y < y1:
    print("S")
elif x > x2:
    print("E")
else:
    print("W")
```

Но у задачи есть и более простое решение. Заметим, что программа должна вывести букву N во всех случаях, когда пловец находится севернее плота, в том числе и в областях NE и NW, то есть при условии $y > y_2$. Аналогично нужно вывести букву S всегда при выполнении условия $y < y_1$, букву W при условии $x < x_1$, букву E при условии $x > x_2$. При этом все «угловые» направления NW, NE, SW, SE получатся автоматически — сначала будет выведена одна из букв N или S, а затем одна из букв W или E. Такое решение содержит всего четыре условные инструкции if.

Пример решения на языке Python

```
x1 = int(input())
y1 = int(input())
x2 = int(input())
y2 = int(input())
x = int(input())
y = int(input())
ans = ""
if y > y2:
    ans += "N"
if y < y1:
    ans += "S"
if x < x1:
    ans += "W"
if x > x2:
    ans += "E"
print(ans)
```

Задача 7. Пакуем чемоданы!

Алёна собирает вещи в отпуск. С собой в самолёт она может взять ручную кладь и багаж. Для ручной клади у Алёны есть рюкзак, а для багажа – огромный чемодан.

По правилам перевозки масса ручной клади не должна превосходить S кг, а багаж может быть любой массы (за сверхнормативный багаж Алёна готова доплатить). Разумеется, наиболее ценные вещи – ноутбук, фотоаппарат, документы и т. д. – Алёна хочет положить в ручную кладь.

Алёна разложила все свои вещи в порядке уменьшения их ценности и начинает складывать наиболее ценные вещи в рюкзак. Она действует следующим образом – берёт самый ценный предмет, и если его масса не превосходит S , то кладёт его в рюкзак, иначе кладёт его в чемодан. Затем она берёт следующий по ценности предмет, если его можно положить в рюкзак, то есть если его масса вместе с массой уже положенных в рюкзак вещей не превосходит S , то кладёт его в рюкзак, иначе в чемодан, и таким же образом процесс продолжается для всех предметов в порядке убывания их ценности.

Определите вес рюкзака и чемодана после того, как Алёна сложит все вещи.

Первая строка входных данных содержит число S – максимально разрешённый вес рюкзака. Во второй строке входных данных записано число N – количество предметов. В следующих N строках даны массы предметов, сами предметы перечислены в порядке убывания ценности (сначала указана масса самого ценного предмета, затем масса второго по ценности предмета и т. д.). Все числа натуральные, число S не превосходит 2×10^9 , сумма весов всех предметов также не превосходит 2×10^9 . Значение N не превосходит 10^5 .

Программа должна вывести два числа – вес рюкзака и вес чемодана (вес пустого рюкзака и чемодана не учитывается).

Пример входных и выходных данных

| Ввод | Вывод | Примечание |
|-----------------------------------|---------|---|
| 20 5 6 10 5 2 3 | 18 8 | Максимально возможная масса рюкзака 20 кг. Дано 5 предметов весом 6, 10, 5, 2, 3. Сначала предмет весом 6 кладётся в рюкзак, затем предмет весом 10 тоже кладётся в рюкзак. Предмет весом 5 нельзя положить в рюкзак, так как тогда вес рюкзака станет 21 кг, поэтому предмет весом 5 кладётся в чемодан. Затем предмет весом 2 кладётся в рюкзак, а предмет весом 3 – в чемодан. Вес рюкзака $6 + 10 + 2 = 18$, вес чемодана $5 + 3 = 8$. |

Система оценивания

Решение, правильно работающее только для случаев, когда все входные числа не превосходят 100, будет оцениваться в 4 балла.

Решение

Эта задача оказалась самой простой для большинства участников. Действительно, в этой задаче нет ни «подводных камней», как в первой задаче, ни объёмного разбора случаев, как во второй задаче. Необходимо просто реализовать тот алгоритм, который описан в условии: завести две переменные для хранения массы рюкзака и массы чемодана, считывать очередное число и прибавлять его либо к массе рюкзака, либо к массе чемодана, следя за тем, чтобы масса чемодана не превышала значения S .

Пример решения на языке Python

```
s = int(input())
n = int(input())
s1 = 0
s2 = 0
for i in range(n):
    a = int(input())
    if s1 + a <= s:
        s1 += a
    else:
        s2 += a
print(s1, s2)
```

Типичная ошибка в этой задаче — использование строгого неравенства $s1 + a < s$ вместо нестрогого $s1 + a <= s$. Такие решения набирали 6 баллов.