**Решения задач**

1. ***Кружки Эйлера*** *(100 баллов)*

Если , то учеников, посещающих оба кружка, может и не найтись, и ответ равен 0. В противном случае можем вычислить «избыток» учеников: . Это то минимальное количество учеников, которые посещают оба кружка, поскольку именно оно не вписывается в общее число *N*.

Пример решения задачи на школьном алгоритмическом языке:

алг

нач

цел N, A, B

ввод N, A, B

если A + B <= N

то

вывод 0

иначе

вывод A + B - N

всё

кон

Пример решения задачи на Pascal:

var

N, A, B: Integer;

begin

Read(N, A, B);

if A + B <= N then

Write(0)

else

Write(A + B - N);

end.

Пример решения задачи на Python:

n, a, b = map(int, input().split())

if a + b <= n:

print(0)

else:

print(a + b - n)

Другой вариант решения на Python:

n, a, b = map(int, input().split())

print(max(0, a + b – n))

1. ***Умный лифт*** (100 баллов)

Вначале найдём порядковый номер нужной квартиры подъезде. Если *N*<*P*, то ответ — 0 (квартира находится в одном из предыдущих подъездов). В противном случае скорректируем номер квартиры. Новый номер квартиры — *M*=*N*–*P*+1.

Определим теперь, на каком этаже находится квартира с порядковым номером *M*. Номер этажа может быть найден целочисленным делением *M* на *K* — число квартир на этаже. При этом, если квариира последняя на этаже, то она нацело делится на *K*, и результат для неё получается на 1 больше, чем нужно. Следует выполнить проверку делимости *M* на *K* и, если *M* делится на *K*, то вычитать из результата 1. Другой вариант — использовать формулу «(M - 1) div K + 1» где div — операция целочисленного деления. При совмещении этой формулы с предыдущей получаем «(N - P) div K + 1».

Пример решения задачи на школьном алгоритмическом языке:

алг

нач

цел N, K, P

ввод N, K, P

если N < P

то

вывод 0

иначе

вывод div(N - P, K) + 1

всё

кон

Пример решения задачи на Pascal:

var

N, K, P: Integer;

begin

Read(N, K, P);

if N < P then

Write(0)

else

Write((N - P) div K + 1);

end.

Пример решения задачи на Python:

n, k, p = map(int, input().split())

if n < p:

print(0)

else:

print((n - p) // k + 1)

Другой вариант решения на Python:

n, k, p = map(int, input().split())

print(0 if n < p else (n - p) // k + 1)

1. ***Стриж*** *(100 баллов)*

Непосредственная проверка всех вариантов составления комбинации затруднена из-за большого числа этих вариантов (24). Поэтому расположим числа в порядке их возрастания (неубывания), то есть отсортируем их. Это позволит ограничиться проверкой только двух случаев:

* для чисел {a, b, c, d} выполняется равенство *a* = *b*–1 = *c*–2;
* для чисел {a, b, c, d} выполняется равенство *b* = *c*–1 = *d*–2.

При выполнении первого равенства нужно выводить *a*, в противном случае проверить второе и, если оно выполняется, то вывести *b*, в противном случае вывести 0.

Сортировку четырёх чисел можно выполнить либо имеющимися в языке программирования методами, либо выполнить серию попарных сравнений и обменов (их будет не более 6).

Пример решения задачи на Python (описанный способ решения):

a, b, c, d = sorted(map(int, input().split()))

if a == b - 1 == c - 2:

print(a)

elif b == c - 1 == d - 2:

print(b)

else:

print(0)

Другой вариант решения на Python (способ решения без сортировки):

s = list(map(int, input().split()))

m = 14

for x in s:

if (x + 1 in s) and (x + 2 in s):

m = min(m, x)

print(0 if m == 14 else m)

Пример решения задачи на Pascal (способ решения без сортировки):

var

i, a, k: Integer;

s: array [1 .. 13] of Integer;

begin

for i := 1 to 13 do

s[i] := 0;

for i := 1 to 4 do

begin

Read(a);

s[a] := s[a] + 1;

end;

k := 0;

for i := 1 to 11 do

if (s[i] > 0) and (s[i + 1] > 0) and (s[i + 2] > 0) then

begin

k := i;

break;

end;

Write(k)

end.

1. ***Каркас (100 баллов)***

В этой задаче необходимо вывести закономерность (формулу), по которой суммарная длина балок зависит от количества рёбер.

*N*

*M*

*K*

Имеются три вида рёбер в зависимости от их направления — в длину, в ширину и в высоту склада.

Подсчитаем количество рёбер, направленных в длину склада. В длину умещается *N* таких рёбер. В ширину выходит (*M* + 1) таких линий, а в высоту — (*K* + 1) таких слоёв. Общее количество рёбер такого типа — *N* · (*M* + 1) · (*K* + 1).

Аналогично подсчитаем количество рёбер, направленных в ширину склада, — *M* · (*N* + 1) · (*K* + 1).

Количество рёбер, направленных в высоту склада, — *K* · (*N* + 1) · (*M* + 1).

Сложим количества рёбер всех трёх видов.

Итоговая формула — .

Другой вид формулы —.

Пример решения задачи на Pascal:

var

N, M, K: Integer;

begin

Read(N, M, K);

Writeln(N\*(M+1)\*(K+1)+M\*(N+1)\*(K+1)+K\*(M+1)\*(N+1));

end.

Пример решения задачи на Python:

n, m, k = map(int, input().split())

print(n\*(m+1)\*(k+1)+m\*(n+1)\*(k+1)+k\*(m+1)\*(n+1))