



ИНФОРМАТИКА

ВАРИАНТ 1

Задача 1 (10 баллов)

На командный пункт артиллеристского дивизиона разведка предоставила зашифрованный участок местности с делением на квадраты.

	A	B	C	D	E
1	0	-1	3	-1	-4
2	2	0	2	0	3
3	1	-6	-5	1	-5
4	8	0	-4	0	9
5	4	5	-4	0	-2

Для дешифровки сообщения в распоряжении командования имеются формулы, указанные ниже.

	G	H	I	J
1	=СЧЁТЕСЛИ(A1:E1;ABS(A1))	=ЕСЛИ(B1;СУММ(A1;E1);МИН(C1;E1))	=СЧЁТЕСЛИ(B1:D1;">"&B1)	=СУММ(G1:I1)
2	=СЧЁТЕСЛИ(A2:E2;ABS(A2))	=ЕСЛИ(B2;СУММ(A2;E2);МИН(C2;E2))	=СЧЁТЕСЛИ(B2:D2;">"&B2)	=СУММ(G2:I2)
3	=СЧЁТЕСЛИ(A3:E3;ABS(A3))	=ЕСЛИ(B3;СУММ(A3;E3);МИН(C3;E3))	=СЧЁТЕСЛИ(B3:D3;">"&B3)	=СУММ(G3:I3)
4	=СЧЁТЕСЛИ(A4:E4;ABS(A4))	=ЕСЛИ(B4;СУММ(A4;E4);МИН(C4;E4))	=СЧЁТЕСЛИ(B4:D4;">"&B4)	=СУММ(G4:I4)
5	=СЧЁТЕСЛИ(A5:E5;ABS(A5))	=ЕСЛИ(B5;СУММ(A5;E5);МИН(C5;E5))	=СЧЁТЕСЛИ(B5:D5;">"&B5)	=СУММ(G5:I5)
6	=СЧЁТЕСЛИ(A1:A5;ABS(A3))	=ЕСЛИ(A2;СУММ(A1;A5);МИН(A2;A4))	=СЧЁТЕСЛИ(A2:A4;">"&A1)	=СУММ(G6:I6)
7	=СЧЁТЕСЛИ(B1:B5;ABS(B3))	=ЕСЛИ(B2;СУММ(B1;B5);МИН(B2;B4))	=СЧЁТЕСЛИ(B2:B4;">"&B1)	=СУММ(G7:I7)
8	=СЧЁТЕСЛИ(C1:C5;ABS(C3))	=ЕСЛИ(C2;СУММ(C1;C5);МИН(C2;C4))	=СЧЁТЕСЛИ(C2:C4;">"&C1)	=СУММ(G8:I8)
9	=СЧЁТЕСЛИ(D1:D5;ABS(D3))	=ЕСЛИ(D2;СУММ(D1;D5);МИН(D2;D4))	=СЧЁТЕСЛИ(D2:D4;">"&D1)	=СУММ(G9:I9)
10	=СЧЁТЕСЛИ(E1:E5;ABS(E3))	=ЕСЛИ(E2;СУММ(E1;E5);МИН(E2;E4))	=СЧЁТЕСЛИ(E2:E4;">"&E1)	=СУММ(G10:I10)

Рассчитайте значения в ячейках от J1 до J10 и определите по ним, в соответствии с таблицей ниже, названия ячеек, в которых находятся опорные пункты противника.

A1	A2	A3	A4	A5	B1	B2	B3	B4	B5	C1	C2	C3
-10	-9	-8	-7	-6	-5	-4	-3	-2	-1	0	1	2
C4	C5	D1	D2	D3	D4	D5	E1	E2	E3	E4	E5	
3	4	5	6	7	8	9	10	11	12	13	14	

В ответе названия ячеек запишите последовательно, слева на право, начиная со значения, соответствующего ячейке J1, затем J2, J3 и т.д.

Решение:



Для решения данной задачи необходимо рассчитать значения в ячейках G1:I10, затем найти значение сумм в колонке J1:J10.

	G	H	I	J	K
1	1	8	1	10	
2	2	6	0	8	
3	1	0	1	2	
4	1	-4	0	-3	
5	0	-10	0	-10	
6	1	0	0	1	
7	1	4	2	7	
8	0	-1	0	-1	
9	2	0	3	5	
10	0	-2	2	0	

На основании значений в колонке J1:10 получаем названия требуемых ячеек.

Ответ:

B4	D1	C1	B3	C4	D4	C3	B5	C5	B2
----	----	----	----	----	----	----	----	----	----

Задача 2 (10 баллов)

Определите четырехзначное число по его описанию Честным человеком и Лгуном. Честный человек всегда говорит правду. Лгун всегда врёт. Ответ поясните.

Честный человек:

- 1) Если от второй цифры числа отнять единицу, то получится третья цифра числа.
- 2) Если разделить первую цифру числа на три, то получится четвертая цифра числа.

Лгун:

- 1) Если от первой цифры числа отнять четвертую цифру числа, то не получится вторая цифра числа.
- 2) Первая цифра числа – не девять.

Решение:

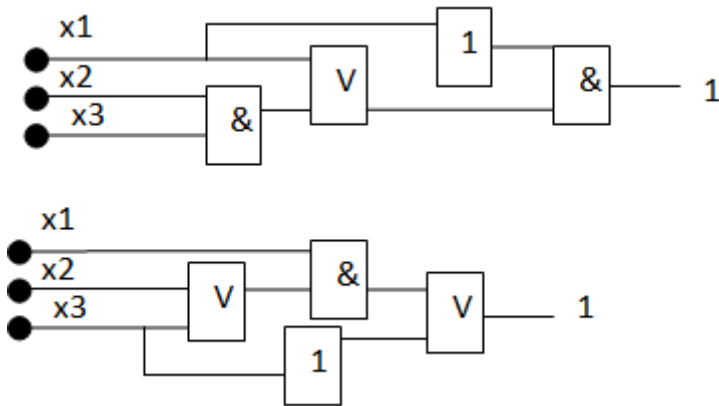
Поскольку Лгун всегда врет, то из его утверждения «Первая цифра числа – не девять», делаем вывод, что первая цифра – 9. Из утверждения Честного человека «Если разделить первую цифру числа на три, то получится четвертая цифра числа», понимаем, что четвертая цифра – 3. Далее из утверждения Лгуна «Если от первой



цифры числа отнять четвёртую цифру числа, то не получится вторая цифра числа», делаем вывод, что вторая цифра равна 6. И третью цифру устанавливаем из утверждения Честного человека «Если от второй цифры числа отнять единицу, то получится третья цифра числа», она получается равной 5. Получаем четырёхзначное число – 9653.

Задача 3 (10 баллов)

Даны логические схемы. Найти все наборы, при которых обе логические схемы возвращают значение истина.



Решение.

Для начала необходимо составить выражения, которые зашифрованы в рамках схемы:

1. Для первой схемы: $(x2 \wedge x3 \vee x1) \wedge \bar{x}1$
2. Для второй схемы: $(x2 \vee x3) \wedge x1 \vee \bar{x}1$

После составления выражений необходимо построить таблицы истинности:

Таблица истинности для первой схемы:

x1	x2	x3	$\bar{x}1$	$x2 \wedge x3$	$x2 \wedge x3 \vee x1$	$(x2 \wedge x3 \vee x1) \wedge \bar{x}1$
0	0	0	1	0	0	0
0	0	1	1	0	0	0
0	1	0	1	0	0	0
0	1	1	1	1	1	1
1	0	0	0	0	1	0
1	0	1	0	0	1	0
1	1	0	0	0	1	0
1	1	1	0	1	1	0

Таблица истинности для второй схемы:

x1	x2	x3	$\bar{x}1$	$x2 \vee x3$	$(x2 \vee x3) \wedge x1$	$(x2 \vee x3) \wedge x1 \vee \bar{x}1$
----	----	----	------------	--------------	--------------------------	--



0	0	0	1	0	0	1
0	0	1	1	1	0	1
0	1	0	1	1	0	1
0	1	1	1	1	0	1
1	0	0	0	0	0	0
1	0	1	0	1	1	1
1	1	0	0	1	1	1
1	1	1	0	1	1	1

Ответ: набор 011.

Задача 4 (20 баллов)

Ваня знает, что IPv4-адрес его компьютера 192.168.56.1. Маска префикс /24. Помогите ему определить адрес сети, адрес хоста, количество свободных узлов и широковещательный адрес.

Решение:

Маска префикс /24 является маской 255.255.255.0.

Чтобы определить адрес сети необходимо перевести IP-v4 адрес 192.168.56.1 и маску 255.255.255.0 в двоичную систему счисления.

192.168.56.1=11000000.10101000.00111000.00000001

255.255.255.0=11111111.11111111.11111111.00000000

Далее необходимо найти результат конъюнкции двоичных значений. В ответе получится 11000000.10101000.00111000.00000000. Переведя в десятичную систему 192.168.56.0 – адрес сети. Адрес хоста в данном случае – 1 в IP-адресе 192.168.56.1

Для того, чтобы получить широковещательный адрес сети необходимо найти количество подсетей. Зная, что в текущей подсети диапазон 0-24 можно определить количество широковещательных адресов по формулам:

- 1). $32-24=8$ – разность бит в адресе и крайнего значения диапазона;
- 2). $2^8=256$ – определение количества подсетей.

Отсюда получается, что широковещательный адрес = 192.168.56.255.

Свободные узлы – это все узлы – 2, т.к. один из адресов зарезервирован под широковещательный адрес, а другой под адрес сети. Соответственно: $2^8-2=254$.

Ответ: 192.168.0 – адрес сети; адрес хоста – 1, 192.168.255 – широковещательный адрес, количество свободных узлов – 254.



Задача 5 (25 баллов)

ДЕНЬ ОЛЕНЕВОДА

В одном из северных городов ежегодно празднуют день оленевода, и люди со всего города собираются вместе, устраивая различные спортивные состязания.

В конце празднования люди любят побаловать себя вкуснейшим оленьим мясом, колбасами и копчеными вырезками. Постоянная участница, Катя съела наверно 20 килограмм различного мяса, но от морозной свежести на улице и согревающих напитков, аппетит становился всё больше.



День оленевода популярный праздник, и в этом году присутствовало N человек, каждому из них выдали бирку с номером от 1 до N . По окончании дня оленевода будет объявлен Оленевод года. Рейтинг составляется в соответствии с общим количеством съеденных килограммов оленины. Люди уже съели довольно много мяса: k -й человек съел $A[k]$ килограммов оленины.

Главный оленевод племени, Барсул уже подготовил огромный олений окорок, который он раздаст людям в соотношении $B[1] : B[2] : \dots : B[N]$, именно в таком порядке, но он еще не знает общее количество (количество килограммов) оленины, которую он будет распределять.

Барсул оказывает непосредственное влияние на этот список, выбирая количество оленины для распределения. Хотя Барсулу много раз предлагали взятки, он каждый раз отказывался, говоря, что он честный человек, который не нарушает своих принципов.

Барсул заботится о порядке, и хочет, чтобы порядок людей был в точной форме $1, 2, 3, \dots, N$, соответственно от того, кто съел больше килограммов оленины, к тем, кто съел меньше, допуская при этом ничьи между участниками. Помогите Барсулу выбрать общее количество оленины, которое он раздаст (в соотношении, указанном ранее), чтобы наградить участников.

Входные данные:

Первая строка ввода содержит целое число N ($2 \leq N \leq 100$) - количество претендентов на премию "Оленевод года".



Олимпиада школьников «Гранит науки» - 2024

Каждая из следующих k строк содержит целые числа $A[k]$ и $B[k]$, как указано в тексте ($0 \leq A[k], B[k] \leq 106$). Хотя бы одно из чисел $B[k]$ не будет равно 0.

Результат должен содержать «-1» и выводиться на экран, если требуемого порядка невозможно достигнуть. В противном случае необходимо вывести количество оленины в килограммах – вещественное число от 0 до 107 (включительно). Если существует несколько возможных решений, выведите любое из них.

Пример:

Ввод: 5 15 4 6 7 12 5 9 6 1 7	Результат: 87
Ввод: 3 2 1 4 0 0 3	Результат: -1

Исходные данные:

5
35 4
36 3
42 2
49 1
21 5

Решение:

Люди от 1 до N получают, соответственно, B_1X , B_2X , ..., B_NX килограммов оленины, учитывая X . Нужно найти X такое, чтобы последовательность была правильной; исходя из этого, легко вычислить общее количество розданных килограммов.



Мы хотим, чтобы следующее было правдой: $A_1 + B_1X \geq A_2 + B_2X \geq \dots \geq A_N + B_NX$.

Разберем первое неравенство, остальные анализируются аналогично.

Из $A_1 + B_1X \geq A_2 + B_2X$ следует $(B_1 - B_2)X \geq A_2 - A_1$.

Мы хотим разделить это неравенство на $B_1 - B_2$, чтобы в левой части остался только X .

Однако нам нужно быть осторожными!

Если $B_1 - B_2 = 0$, то мы не можем выполнить деление, но неравенство превращается в $0 \geq A_2 - A_1$. В этом случае, если $A_2 - A_1 > 0$, мы выводим -1 , так как это противоречие.

В противном случае неравенство верно, и мы переходим к следующему.

Если $B_1 - B_2 \neq 0$, мы можем выполнить деление, но деление на отрицательное число меняет знаки неравенства, поэтому мы получим либо $X \geq (A_2 - A_1) / (B_1 - B_2)$, либо $X \leq (A_2 - A_1) / (B_1 - B_2)$. В одном случае мы получили нижнюю, а в другом - верхнюю границу для X .

Повторив эту операцию для всех $N - 1$ неравенств, мы получили некоторые нижние и верхние границы для X . Из всех нижних границ нас интересует только самая наибольшая, а из всех верхних границ нас интересует только наименьшая. Если между этими двумя границами есть число, то есть если нижняя граница меньше или равна верхней, то решение существует: мы можем использовать среднее значение нижней и верхней в качестве X . В противном случае решения не существует.

Задача также решается с помощью двоичного поиска;

Реализация на C++

```
#include <algorithm>
```

```
#include <cstdio>
```

```
#include <iostream>
```

```
using namespace std;
```

```
const int NN = 1000000;
```

```
const double GG = 1000000;
```

```
int a[NN], b[NN];
```

```
int main () {
```

```
    int n;
```

```
    scanf("%d", &n);
```



```
int sa = 0;
for (int i = 0; i < n; ++i) {
    scanf("%d%d", a + i, b + i);
    sa += b[i];
}
double dg = 0, gg = GG;
for (int i = 1; i < n; ++i) {
    int rb = b[i - 1] - b[i];
    int ra = a[i] - a[i - 1];
    if (rb == 0 && ra > 0) {
        printf("-1\n");
        return 0;
    }
    if (rb > 0) dg = max(dg, (double)ra / (double)rb);
    if (rb < 0) gg = min(gg, (double)ra / (double)rb);
}
if ((dg < gg + 1e-9))
    printf("%.12lf\n", sa * (dg + gg) / 2);
else
    printf("-1\n");
}
```

Реализация на Python

```
n = int(input())
a = [0] * n
b = [0] * n
sa = 0
for i in range(n):
    a[i], b[i] = map(int, input().split())
    sa += b[i]

dg = 0
```




```
gg = 1000000
```

```
for i in range(1, n):  
    rb = b[i - 1] - b[i]  
    ra = a[i] - a[i - 1]  
    if rb == 0 and ra > 0:  
        print("-1")  
        exit()  
    if rb > 0:  
        dg = max(dg, ra / rb)  
    if rb < 0:  
        gg = min(gg, ra / rb)  
  
if dg < gg + 1e-9:  
    print("{:.12f}".format(sa * (dg + gg) / 2))  
else:  
    print("-1")
```

Демонстрация работы на C++



```
Granit2024 (Глобальная область)
1 #define _CRT_SECURE_NO_WARNINGS
2
3 #include <algorithm>
4 #include <cstdio>
5 #include <iostream>
6 using namespace std;
7
8 const int NN = 1000000;
9 const double GG = 1000000;
10
11 int a[NN], b[NN];
12
13 int main() {
14     int n;
15     scanf("%d", &n);
16     int sa = 0;
17     for (int i = 0; i < n; ++i) {
18         scanf("%d%d", a + i, b + i);
19         sa += b[i];
20     }
21     double dg = 0, gg = GG;
22     for (int i = 1; i < n; ++i) {
23         int rb = b[i - 1] - b[i];
24         int ra = a[i] - a[i - 1];
25         if (rb == 0 && ra > 0) {
26             printf("-1\n");
27             return 0;
28         }
29         if (rb > 0) dg = max(dg, (double)ra);
30         if (rb < 0) gg = min(gg, (double)ra);
31     }
32     if ((dg < gg + 1e-9))
33         printf("%.2lf\n", sa * (dg + gg) / 2);
34     else
35         printf("-1\n");
36 }
37
```

Консоль отладки Microsoft V

```
5 Показать выходные данные из: Сборка
35 4 Сборка начата в 23:51...
36 3 --- Сборка начата: проект: Granit2024, Конфигурация: De
37 granit2024.cpp
42 2 granit2024.vcxproj -> E:\CPP\Granit2024\Granit2024\x64\Del
49 1 ===== Сборка: успешно выполнено - 1, со сбоям - 0, в а
21 5 ===== Сборка завершено в 23:51 и заняло 01,063 с =====
105.00
```

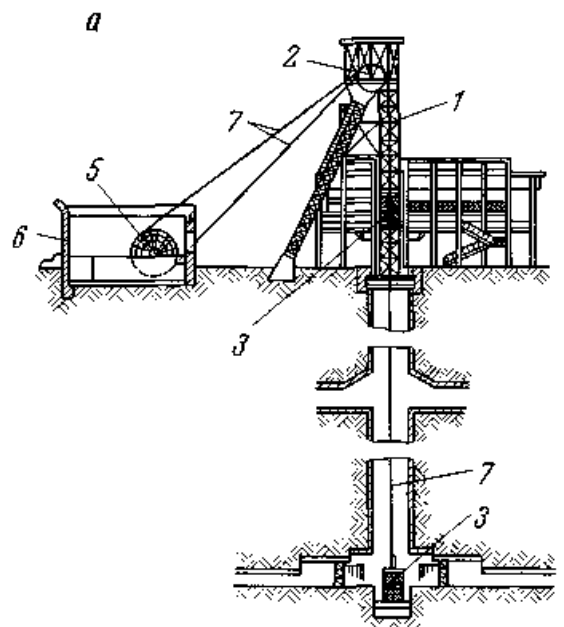
Ответ:

105.0

Задача 6 (25 баллов)

АВАРИЯ НА ШАХТЕ

В одной из горнодобывающих компаний, после произошедшей аварии на шахте руководство озаботилось повышением безопасности и уменьшением времени эвакуации людей в случае аварийной





ситуации. На картинке представлена клетьевая подъемная установка, которая выполняет операции по спуску-подъему людей, спуску материалов и подъему горной породы.

Перед компанией производителей подъемной установки была поставлена задача определить минимальное время за которое подъемная установка сможет эвакуировать шахтеров со всех уровней разрабатываемого месторождения. Известно, что на k -м уровне работает $p[k]$ шахтеров, а безопасная зона (поверхность) на нулевом этаже. Изначально клеть подъемной установки, вместимостью 8 человек, расположена на поверхности. Клеть за 10 секунд проходит один этаж, а посадка и высадка происходит мгновенно.

Разработайте программу, которая определяет минимальное время эвакуации шахтеров со всех уровней месторождения.

Входные значения: первая строка одно число (количество подземных уровней шахты). В следующих строках вам даются целые числа – количество шахтеров, начиная с первого уровня.

Результат должен содержать одно целое число – минимальное время эвакуации всех шахтеров.

Несмотря на то, что в вашем задании указаны достаточно малые значения, ваша программа должна выполняться и для больших входных значений.

Пример

Ввод:	Результат:
Для клетки вместимостью 2 чел.	120
3	
1	
0	
3	

Исходные данные:

10

18

3

12

13



6

10

8

1

12

3

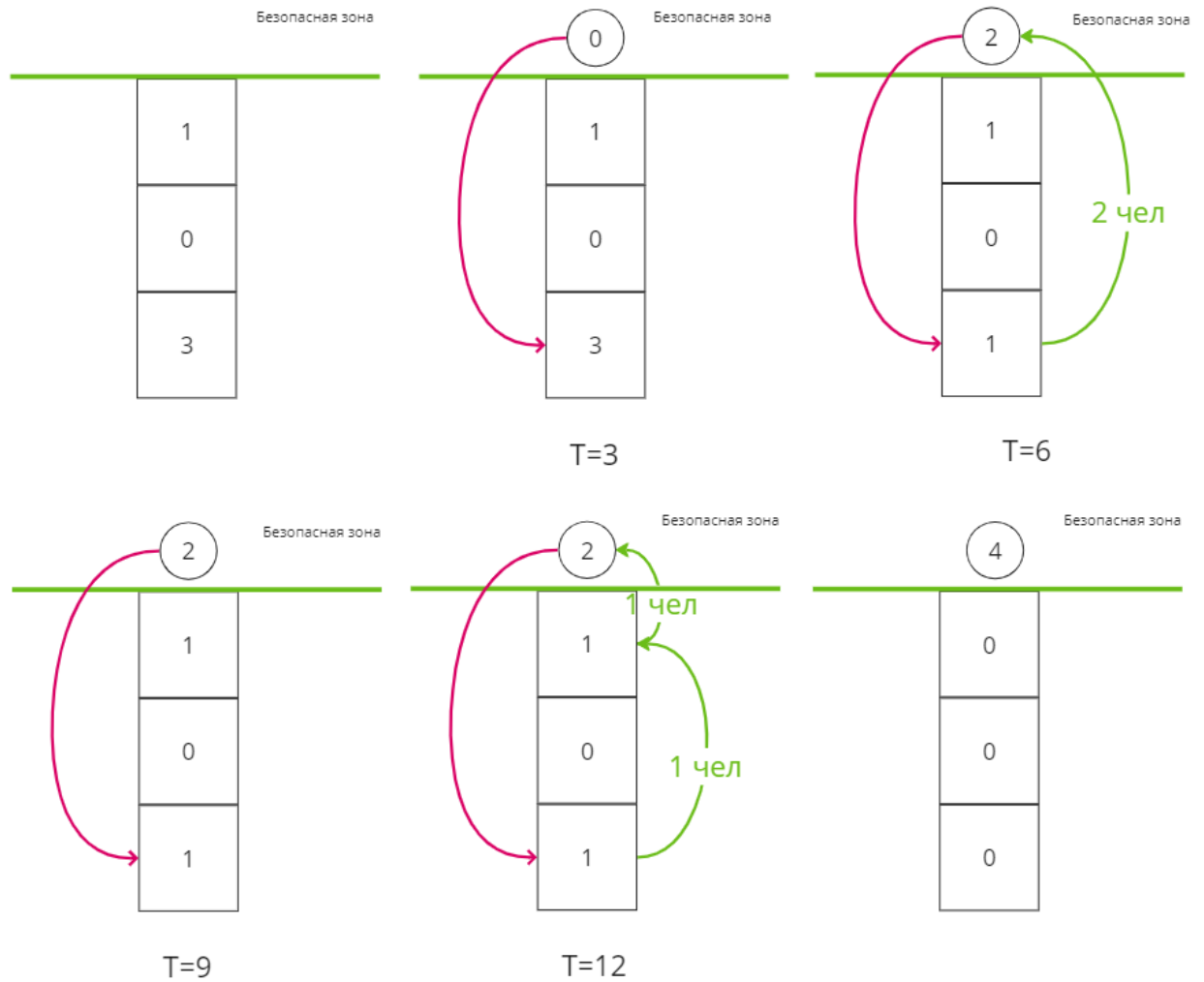
Решение:

Сначала едем на самый глубокий уровень, и забираем всех кто вместиться, если, к примеру, там 1 человек, а вместимость клетки 4 человека, то заполняем свободные места на следующих уровнях.

Оптимизации:

Если на уровне 1000000 человек, а вместимость клетки 1 человек, то можно посчитать количество поездок на этот уровень математически.

Так же если мы забрали всех шахтеров с этажа то эту информацию нужно хранить, причем по пути вверх можно проверять есть ли шахтеры не последующих уровнях, чтобы во время следующего спуска попасть на новый самый глубокий уровень с людьми.



Для удобства расчет сначала ведется как будто прохождение уровней проходит за 1 секунду, а в конце результат умножается на количество секунд описанные в задании.

Реализация на C++

```
#include <iostream>
```

```
#include <vector>
```

```
using namespace std;
```

```
int elevator(int capacity, int floors, vector<int>& people) {
```

```
    int flag = 0;
```

```
    int ans = 0;
```

```
    int k_now = 0;
```

```
    for (int i = floors - 1; i >= 0; i--) {
```

```
        if (people[i] != 0) {
```

```
            if (flag == 0) {
```



```
    ans += i + 1;
    flag = 1;
}

people[i] += k_now;

if (people[i] / capacity > 0 && people[i] % capacity == 0) {
    ans += 2 * (people[i] / capacity - 1) * (i + 1) + (i + 1);
    flag = 0;
    k_now = 0;
}
else if (people[i] / capacity > 0 && people[i] % capacity != 0) {
    k_now = people[i] % capacity;
    ans += 2 * (people[i] / capacity) * (i + 1) + 1;
}
else {
    k_now = people[i];
    ans += 1;
}
}
else {
    if (flag == 1) {
        ans += 1;
    }
}
}
return ans;
}

int main() {
    int capacity = 8;
    int time_level = 10;
    int floors;
```



```
cin >> floors;
vector<int> people(floors);
for (int i = 0; i < floors; i++) {
    cin >> people[i];
}
cout << elevator(capacity, floors, people) * time_level << endl;
return 0;
}
```

Реализация на Python

```
def elevator(capacity, floors, people):
    flag = 0
    ans = 0
    k_now = 0
    for i in range(floors - 1, -1, -1):
        if people[i] != 0:
            if flag == 0:
                ans += i + 1
                flag = 1

            people[i] += k_now

            if people[i] // capacity > 0 and people[i] % capacity == 0:
                ans += 2 * (people[i] // capacity - 1) * (i + 1) + (i + 1)
                flag = 0
                k_now = 0
            elif people[i] // capacity > 0 and people[i] % capacity != 0:
                k_now = people[i] % capacity
                ans += 2 * (people[i] // capacity) * (i + 1) + 1
            else:
                k_now = people[i]
                ans += 1
            else:
```



```
if flag == 1:  
    ans += 1  
return ans
```

```
capacity = 8  
time_level = 10  
floors = int(input())  
people = [int(input()) for i in range(floors)]  
print(elevator(capacity, floors, people) * time_level)
```

Демонстрация на C++

```
Granit2024.cpp  x  
Granit2024 (Глобальная область) main()  
31  
32     }  
33     else {  
34         if (flag == 1) {  
35             ans += 1;  
36         }  
37     }  
38     return ans;  
39 }  
40  
41 int main() {  
42     int capacity = 8; ←  
43     int time_level = 10; ←  
44     int floors;  
45     cin >> floors;  
46     vector<int> people(floors);  
47     for (int i = 0; i < floors; i++) {  
48  
49  
50     cout << elevator(capacity, floors, people) * time_level << endl;  
51     return 0;  
52  
10  
18  
3  
12  
13  
6  
10  
8  
1  
12  
3  
1080
```

Ответ: 1080



ИНФОРМАТИКА
ВАРИАНТ 2

Задача 1 (10 баллов)

На командный пункт артиллеристского дивизиона разведка предоставила зашифрованный участок местности с делением на квадраты.

	A	B	C	D	E
1	6	-1	4	-1	2
2	3	2	-3	0	3
3	5	-2	-5	1	-5
4	-3	0	-4	0	3
5	-6	5	-5	1	-4

Для дешифровки сообщения в распоряжении командования имеются формулы, указанные ниже.

	G	H	I	J
1	=СЧЁТЕСЛИ(A1:E1;ABS(A1))	=ЕСЛИ(B1;СУММ(A1;E1);МИН(C1;E1))	=СЧЁТЕСЛИ(B1:D1;">"&B1)	=СУММ(G1:I1)
2	=СЧЁТЕСЛИ(A2:E2;ABS(A2))	=ЕСЛИ(B2;СУММ(A2;E2);МИН(C2;E2))	=СЧЁТЕСЛИ(B2:D2;">"&B2)	=СУММ(G2:I2)
3	=СЧЁТЕСЛИ(A3:E3;ABS(A3))	=ЕСЛИ(B3;СУММ(A3;E3);МИН(C3;E3))	=СЧЁТЕСЛИ(B3:D3;">"&B3)	=СУММ(G3:I3)
4	=СЧЁТЕСЛИ(A4:E4;ABS(A4))	=ЕСЛИ(B4;СУММ(A4;E4);МИН(C4;E4))	=СЧЁТЕСЛИ(B4:D4;">"&B4)	=СУММ(G4:I4)
5	=СЧЁТЕСЛИ(A5:E5;ABS(A5))	=ЕСЛИ(B5;СУММ(A5;E5);МИН(C5;E5))	=СЧЁТЕСЛИ(B5:D5;">"&B5)	=СУММ(G5:I5)
6	=СЧЁТЕСЛИ(A1:A5;ABS(A3))	=ЕСЛИ(A2;СУММ(A1;A5);МИН(A2;A4))	=СЧЁТЕСЛИ(A2:A4;">"&A1)	=СУММ(G6:I6)
7	=СЧЁТЕСЛИ(B1:B5;ABS(B3))	=ЕСЛИ(B2;СУММ(B1;B5);МИН(B2;B4))	=СЧЁТЕСЛИ(B2:B4;">"&B1)	=СУММ(G7:I7)
8	=СЧЁТЕСЛИ(C1:C5;ABS(C3))	=ЕСЛИ(C2;СУММ(C1;C5);МИН(C2;C4))	=СЧЁТЕСЛИ(C2:C4;">"&C1)	=СУММ(G8:I8)
9	=СЧЁТЕСЛИ(D1:D5;ABS(D3))	=ЕСЛИ(D2;СУММ(D1;D5);МИН(D2;D4))	=СЧЁТЕСЛИ(D2:D4;">"&D1)	=СУММ(G9:I9)
10	=СЧЁТЕСЛИ(E1:E5;ABS(E3))	=ЕСЛИ(E2;СУММ(E1;E5);МИН(E2;E4))	=СЧЁТЕСЛИ(E2:E4;">"&E1)	=СУММ(G10:I10)

Рассчитайте значения в ячейках от J1 до J10 и определите по ним, в соответствии с таблицей ниже, названия ячеек, в которых находятся опорные пункты противника.

A1	A2	A3	A4	A5	B1	B2	B3	B4	B5	C1	C2	C3
-10	-9	-8	-7	-6	-5	-4	-3	-2	-1	0	1	2
C4	C5	D1	D2	D3	D4	D5	E1	E2	E3	E4	E5	
3	4	5	6	7	8	9	10	11	12	13	14	

В ответе названия ячеек запишите последовательно, слева на право, начиная со значения, соответствующего ячейке J1, затем J2, J3 и т.д.



Решение:

Для решения данной задачи необходимо рассчитать значения в ячейках G1:I10, затем найти значение сумм в колонке J1:J10.

	G	H	I	J	K
1	1	8	1	10	
2	2	6	0	8	
3	1	0	1	2	
4	1	-4	0	-3	
5	0	-10	0	-10	
6	1	0	0	1	
7	1	4	2	7	
8	0	-1	0	-1	
9	2	0	3	5	
10	0	-2	2	0	

На основании значений в колонке J1:10 получаем названия требуемых ячеек.

Ответ:

E1	D4	C3	B3	A1	C2	D3	B5	D1	C1
----	----	----	----	----	----	----	----	----	----

Задача 2 (10 баллов)

Определите четырехзначное число по его описанию Честным человеком и Лгуном. Честный человек всегда говорит правду. Лгун всегда врёт. Ответ поясните.

Честный человек:

- 1) Если от третьей цифры числа отнять единицу, то получится первая цифра числа.
- 2) Если вторую цифру числа умножить на два, то получится третья цифра числа.

Лгун:

- 1) Если третью цифру числа разделить на четыре, то не получится четвёртая цифра числа.
- 2) Вторая цифра числа – не четыре.

Решение:

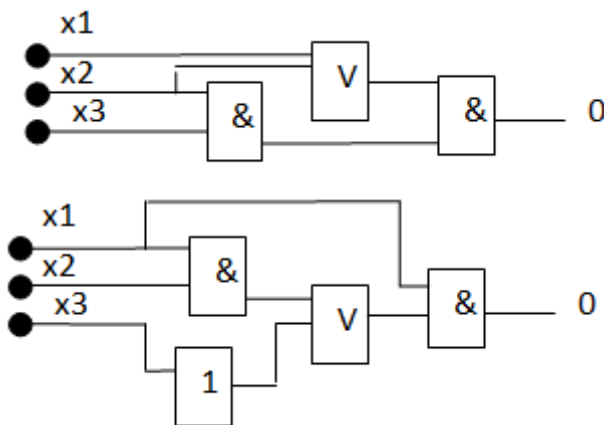
Поскольку Лгун всегда врёт, то из его утверждения «Вторая цифра числа – не четыре», делаем вывод, что вторая цифра – 4. Из утверждения Честного человека «Если вторую цифру числа умножить на два, то получится третья цифра числа», понимаем, что третья цифра – 8. Далее из утверждения Лгуна «Если третью цифру числа разделить на четыре, то не получится четвёртая цифра числа», делаем вывод, что четвёртая цифра



равна 2. И первую цифру устанавливаем из утверждения Честного человека «Если от третьей цифры числа отнять единицу, то получится первая цифра числа», она получается равной 7. Получаем четырёхзначное число – 7482.

Задача 3 (10 баллов)

Даны логические схемы. Найти все наборы, при которых обе логические схемы возвращают значение ложь.



Решение.

Для начала необходимо составить выражения, которые зашифрованы в рамках схемы:

1. Для первой схемы: $(x2 \wedge x3) \wedge (x2 \vee x1)$
2. Для второй схемы: $(x1 \wedge x2 \vee \bar{x}3) \wedge x1$

После составления выражений необходимо построить таблицы истинности:

Таблица истинности для первой схемы:

x1	x2	x3	$x2 \wedge x3$	$x2 \vee x1$	$(x2 \wedge x3) \wedge (x2 \vee x1)$
0	0	0	0	0	0
0	0	1	0	0	0
0	1	0	0	1	0
0	1	1	1	1	1
1	0	0	0	1	0
1	0	1	0	1	0
1	1	0	0	1	0
1	1	1	1	1	1

Таблица истинности для второй схемы:

x1	x2	x3	$\bar{x}3$	$x1 \wedge x2$	$x1 \wedge x2 \vee \bar{x}3$	$(x1 \wedge x2 \vee \bar{x}3) \wedge x1$
0	0	0	1	0	1	0



0	0	1	0	0	0	0
0	1	0	1	0	1	0
0	1	1	0	0	0	0
1	0	0	1	0	1	1
1	0	1	0	0	0	0
1	1	0	1	1	1	1
1	1	1	0	1	1	1

Ответ: 4 набора значений.

Задача 4 (20 баллов)

Ваня знает, что IPv4-адрес его компьютера 192.168.0.107. Маска префикс /24. Помогите ему определить адрес сети, адрес хоста, количество свободных узлов и широковещательный адрес.

Решение:

Маска префикс /24 является маской 255.255.255.0.

Чтобы определить адрес сети необходимо перевести IP-v4 адрес 192.168.0.107 и маску 255.255.255.0 в двоичную систему счисления.

$192.168.0.107 = 11000000.10101000.00000000.01101011$

$255.255.255.0 = 11111111.11111111.11111111.00000000$

Далее необходимо найти результат конъюнкции двоичных значений. В ответе получится 11000000.10101000.00000000.00000000. Переведя в десятичную систему 192.168.0.0 – адрес сети. Адрес хоста – **107**, т.к. ip-адрес заканчивается на 107.

Для того, чтобы получить широковещательный адрес сети необходимо найти количество подсетей. Зная, что в текущей подсети диапазон 0-24 можно определить количество широковещательных адресов по формулам:

- 1). $32-24=8$ – разность бит в адресе и крайнего значения диапазона;
- 2). $2^8=256$ – определение количества подсетей.

Отсюда получается, что широковещательный адрес = 192.168.0.255.

Свободные узлы – это все узлы – 2, т.к. один из адресов зарезервирован под широковещательный адрес, а другой под адрес сети. Соответственно: $2^8-2=254$.

Ответ: 192.0.0 – адрес сети; 107 – адрес хоста, 192.0.0 – широковещательный адрес, количество свободных узлов – 254.



Задача 5 (25 баллов)

ДЕНЬ ОЛЕНЕВОДА

В одном из северных городов ежегодно празднуют день оленевода, и люди со всего города собираются вместе, устраивая различные спортивные состязания.

В конце празднования люди любят побаловать себя вкуснейшим оленьим мясом, колбасами и копчеными вырезками. Постоянная участница, Катя съела наверно 20 килограмм различного мяса, но от морозной свежести на улице и согревающих напитков, аппетит становился всё больше.



День оленевода популярный праздник, и в этом году присутствовало N человек, каждому из них выдали бирку с номером от 1 до N . По окончании дня оленевода будет объявлен Оленевод года. Рейтинг составляется в соответствии с общим количеством съеденных килограммов оленины. Люди уже съели довольно много мяса: k -й человек съел $A[k]$ килограммов оленины.

Главный оленевод племени, Барсул уже подготовил огромный олений окорок, который он раздаст людям в соотношении $B[1] : B[2] : \dots : B[N]$, именно в таком порядке, но он еще не знает общее количество (количество килограммов) оленины, которую он будет распределять.

Барсул оказывает непосредственное влияние на этот список, выбирая количество оленины для распределения. Хотя Барсулу много раз предлагали взятки, он каждый раз отказывался, говоря, что он честный человек, который не нарушает своих принципов.

Барсул заботится о порядке, и хочет, чтобы порядок людей был в точной форме $1, 2, 3, \dots, N$, соответственно от того, кто съел больше килограммов оленины, к тем, кто съел меньше, допуская при этом ничьи между участниками. Помогите Барсулу выбрать общее количество оленины, которое он раздаст (в соотношении, указанном ранее), чтобы наградить участников.

Входные данные: Первая строка ввода содержит целое число N ($2 \leq N \leq 100$) - количество претендентов на премию "Оленевод года".

Каждая из следующих k строк содержит целые числа $A[k]$ и $B[k]$, как указано в тексте ($0 \leq A[k], B[k] \leq 106$). Хотя бы одно из чисел $B[k]$ не будет равно 0.



Олимпиада школьников «Гранит науки» - 2024

Результат должен содержать «-1» и выводиться на экран, если требуемого порядка невозможно достигнуть. В противном случае необходимо вывести количество оленины в килограммах – вещественное число от 0 до 107 (включительно). Если существует несколько возможных решений, выведите любое из них.

Пример:

Ввод: 5 15 4 6 7 12 5 9 6 1 7	Результат: 87
Ввод: 3 2 1 4 0 0 3	Результат: -1

Исходные данные:

5
45 2
22 5
8 7
14 6
41 2



Решение:

Люди от 1 до N получают, соответственно, V_1X , V_2X , ..., V_NX килограммов оленины, учитывая X . Нужно найти X такое, чтобы последовательность была правильной; исходя из этого, легко вычислить общее количество розданных килограммов.

Мы хотим, чтобы следующее было правдой: $A_1 + V_1X \geq A_2 + V_2X \geq \dots \geq A_N + V_NX$.

Разберем первое неравенство, остальные анализируются аналогично.

Из $A_1 + V_1X \geq A_2 + V_2X$ следует $(V_1 - V_2)X \geq A_2 - A_1$.

Мы хотим разделить это неравенство на $V_1 - V_2$, чтобы в левой части остался только X .

Однако нам нужно быть осторожными!

Если $V_1 - V_2 = 0$, то мы не можем выполнить деление, но неравенство превращается в $0 \geq A_2 - A_1$. В этом случае, если $A_2 - A_1 > 0$, мы выводим -1, так как это противоречие.

В противном случае неравенство верно, и мы переходим к следующему.

Если $V_1 - V_2 \neq 0$, мы можем выполнить деление, но деление на отрицательное число меняет знаки неравенства, поэтому мы получим либо $X \geq (A_2 - A_1) / (V_1 - V_2)$, либо $X \leq (A_2 - A_1) / (V_1 - V_2)$. В одном случае мы получили нижнюю, а в другом - верхнюю границу для X .

Повторив эту операцию для всех $N - 1$ неравенств, мы получили некоторые нижние и верхние границы для X . Из всех нижних границ нас интересует только самая наибольшая, а из всех верхних границ нас интересует только наименьшая. Если между этими двумя границами есть число, то есть если нижняя граница меньше или равна верхней, то решение существует: мы можем использовать среднее значение нижней и верхней в качестве X . В противном случае решения не существует.

Задача также решается с помощью двоичного поиска;

Реализация на C++

```
#include <algorithm>
```

```
#include <cstdio>
```

```
#include <iostream>
```

```
using namespace std;
```

```
const int NN = 1000000;
```

```
const double GG = 1000000;
```

```
int a[NN], b[NN];
```



```
int main () {
    int n;
    scanf("%d", &n);
    int sa = 0;
    for (int i = 0; i < n; ++i) {
        scanf("%d%d", a + i, b + i);
        sa += b[i];
    }
    double dg = 0, gg = GG;
    for (int i = 1; i < n; ++i) {
        int rb = b[i - 1] - b[i];
        int ra = a[i] - a[i - 1];
        if (rb == 0 && ra > 0) {
            printf("-1\n");
            return 0;
        }
        if (rb > 0) dg = max(dg, (double)ra / (double)rb);
        if (rb < 0) gg = min(gg, (double)ra / (double)rb);
    }
    if ((dg < gg + 1e-9))
        printf("%.12lf\n", sa * (dg + gg) / 2);
    else
        printf("-1\n");
}
```

Реализация на Python

```
n = int(input())
a = [0] * n
b = [0] * n
sa = 0
for i in range(n):
```




```
a[i], b[i] = map(int, input().split())
sa += b[i]

dg = 0
gg = 1000000

for i in range(1, n):
    rb = b[i - 1] - b[i]
    ra = a[i] - a[i - 1]
    if rb == 0 and ra > 0:
        print("-1")
        exit()
    if rb > 0:
        dg = max(dg, ra / rb)
    if rb < 0:
        gg = min(gg, ra / rb)

if dg < gg + 1e-9:
    print("{:.12f}".format(sa * (dg + gg) / 2))
else:
    print("-1")
```

Демонстрация работы на C++



```
Granit2024 (Глобальная область)
1  #define _CRT_SECURE_NO_WARNINGS
2
3  #include <algorithm>
4  #include <cstdio>
5  #include <iostream>
6  using namespace std;
7
8  const int NN = 1000000;
9  const double GG = 1000000;
10
11 int a[NN], b[NN];
12
13 int main() {
14     int n;
15     scanf("%d", &n);
16     int sa = 0;
17     for (int i = 0; i < n; ++i) {
18         scanf("%d%d", a + i, b + i);
19         sa += b[i];
20     }
21     double dg = 0, gg = GG;
22     for (int i = 1; i < n; ++i) {
23         int rb = b[i - 1] - b[i];
24         int ra = a[i] - a[i - 1];
25         if (rb == 0 && ra > 0) {
26             printf("-1\n");
27             return 0;
28         }
29         if (rb > 0) dg = max(dg, (double)ra);
30         if (rb < 0) gg = min(gg, (double)ra);
31     }
32     if ((dg < gg + 1e-9))
33         printf("%.2lf\n", sa * (dg + gg) / 2);
34     else
35         printf("-1\n");
36 }
37
```

Консоль отладки Microsoft V

5 Показать выходные данные из: Сборка

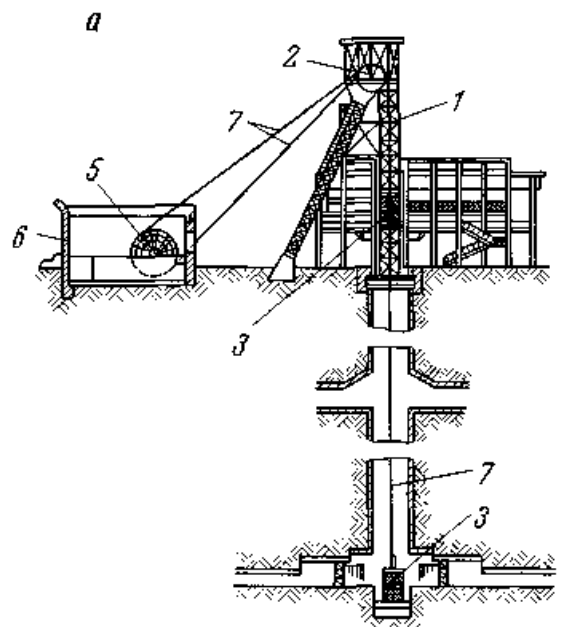
45 2
22 5
8 7
14 6
41 2
151.25

Ответ: 151.25

Задача 6.

АВАРИЯ НА ШАХТЕ

В одной из горнодобывающих компаний, после произошедшей аварии на шахте руководство озаботилось повышением безопасности и уменьшением времени эвакуации людей в случае аварийной





ситуации. На картинке представлена клетьевая подъемная установка, которая выполняет операции по спуску-подъему людей, спуску материалов и подъему горной породы.

Перед компанией производителей подъемной установки была поставлена задача определить минимальное время за которое подъемная установка сможет эвакуировать шахтеров со всех уровней разрабатываемого месторождения. Известно, что на k -м уровне работает $p[k]$ шахтеров, а безопасная зона (поверхность) на нулевом этаже. Изначально клеть подъемной установки, вместимостью 7 человек, расположена на поверхности. Клеть за 7 секунд проходит один этаж, а посадка и высадка происходит мгновенно.

Разработайте программу, которая определяет минимальное время эвакуации шахтеров со всех уровней месторождения.

Входные значения: первая строка одно число (количество подземных уровней шахты). В следующих строках вам даются целые числа – количество шахтеров, начиная с первого уровня.

Результат должен содержать одно целое число – минимальное время эвакуации всех шахтеров.

Несмотря на то, что в вашем задании указаны достаточно малые значения, ваша программа должна выполняться и для больших входных значений.

Пример

Ввод:	Результат:
Для клетки вместимостью 2 чел.	56
3	
3	
0	
1	

Исходные данные:

- 10
- 11
- 3
- 5
- 12



1

10

8

1

0

7

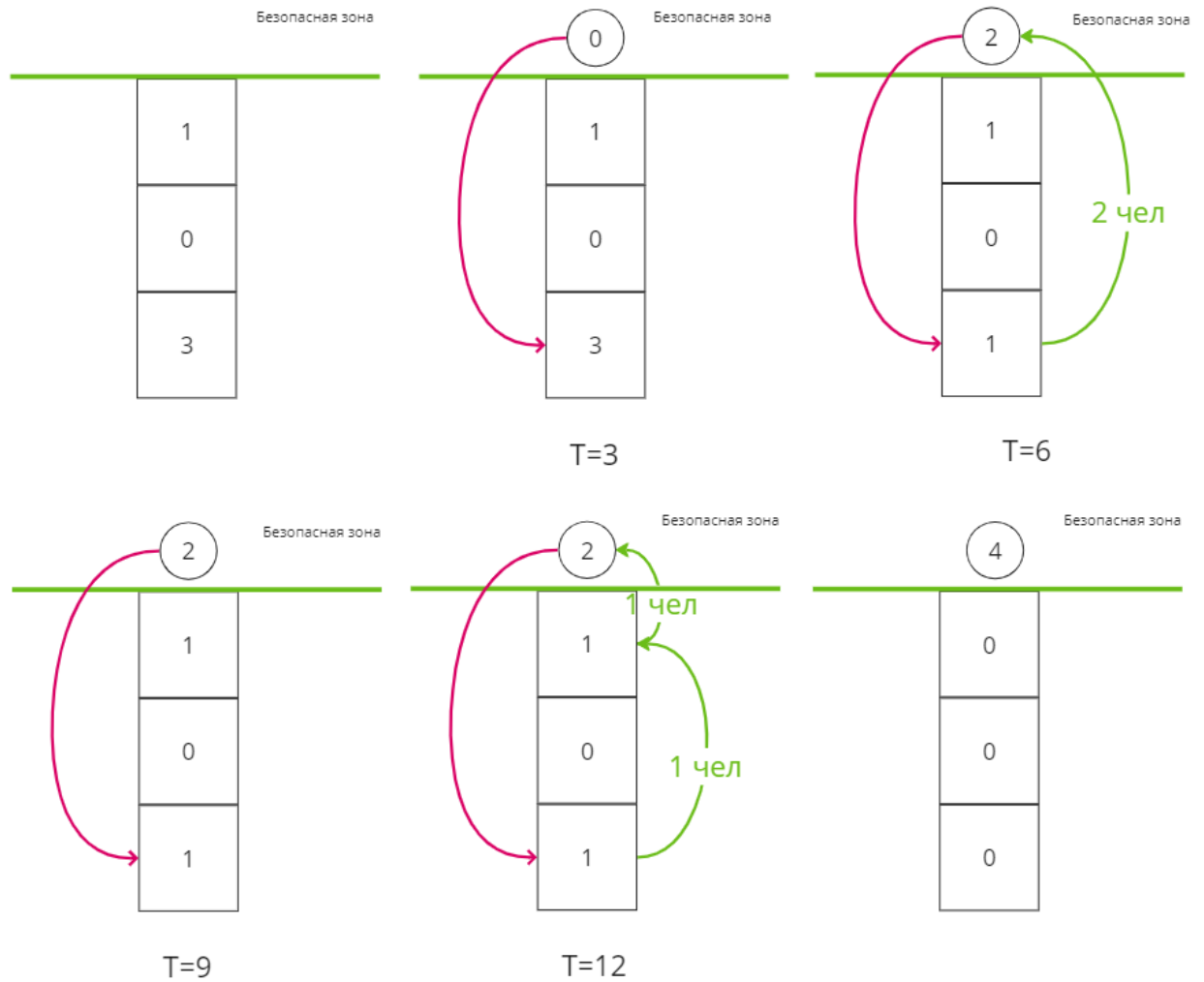
АВАРИЯ НА ШАХТЕ

Сначала едем на самый глубокий уровень, и забираем всех кто вместиться, если, к примеру, там 1 человек, а вместимость клетки 4 человека, то заполняем свободные места на следующих уровнях.

Оптимизации:

Если на уровне 1000000 человек, а вместимость клетки 1 человек, то можно посчитать количество поездок на этот уровень математически.

Так же если мы забрали всех шахтеров с этажа то эту информацию нужно хранить, причем по пути вверх можно проверять есть ли шахтеры не последующих уровнях, чтобы во время следующего спуска попасть на новый самый глубокий уровень с людьми.



Для удобства расчет сначала ведется как будто прохождение уровней проходит за 1 секунду, а в конце результат умножается на количество секунд описанные в задании.

Реализация на C++

```
#include <iostream>
```

```
#include <vector>
```

```
using namespace std;
```

```
int elevator(int capacity, int floors, vector<int>& people) {
```

```
    int flag = 0;
```

```
    int ans = 0;
```

```
    int k_now = 0;
```

```
    for (int i = floors - 1; i >= 0; i--) {
```



```
if (people[i] != 0) {
    if (flag == 0) {
        ans += i + 1;
        flag = 1;
    }

    people[i] += k_now;

    if (people[i] / capacity > 0 && people[i] % capacity == 0) {
        ans += 2 * (people[i] / capacity - 1) * (i + 1) + (i + 1);
        flag = 0;
        k_now = 0;
    }
    else if (people[i] / capacity > 0 && people[i] % capacity != 0) {
        k_now = people[i] % capacity;
        ans += 2 * (people[i] / capacity) * (i + 1) + 1;
    }
    else {
        k_now = people[i];
        ans += 1;
    }
}
else {
    if (flag == 1) {
        ans += 1;
    }
}
return ans;
}

int main() {
    int capacity = 8;
```



```
int time_level = 10;
int floors;
cin >> floors;
vector<int> people(floors);
for (int i = 0; i < floors; i++) {
    cin >> people[i];
}
cout << elevator(capacity, floors, people) * time_level << endl;
return 0;
}
```

Реализация на Python

```
def elevator(capacity, floors, people):
    flag = 0
    ans = 0
    k_now = 0
    for i in range(floors - 1, -1, -1):
        if people[i] != 0:
            if flag == 0:
                ans += i + 1
                flag = 1

            people[i] += k_now

        if people[i] // capacity > 0 and people[i] % capacity == 0:
            ans += 2 * (people[i] // capacity - 1) * (i + 1) + (i + 1)
            flag = 0
            k_now = 0
        elif people[i] // capacity > 0 and people[i] % capacity != 0:
            k_now = people[i] % capacity
            ans += 2 * (people[i] // capacity) * (i + 1) + 1
        else:
            k_now = people[i]
```



```
ans += 1
else:
    if flag == 1:
        ans += 1
return ans
```

```
capacity = 8
time_level = 10
floors = int(input())
people = [int(input()) for i in range(floors)]
print(elevator(capacity, floors, people) * time_level)
```

Демонстрация на C++

```
31
32
33     else {
34         if (flag == 1) {
35             ans += 1;
36         }
37     }
38     return ans;
39 }
40
41 int main() {
42     int capacity = 7;
43     int time_level = 7;
44     int floors;
45     cin >> floors;
46     vector<int> people(floors);
47     for (int i = 0; i < floors; i++) {
48         cin >> people[i];
49     }
50 }
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
```

Ответ: 616



ИНФОРМАТИКА

ВАРИАНТ 3

Задача 1 (10 баллов)

Перед вами зашифрованная карта минных полей.

	A	B	C	D	E
1		=2>2	-5	A1	3
2	-5	-1	-5	-4	=E4+2
3	1	0	-8	C2>5	D2
4	2	-4	-4	0	=D3>3
5	4	1	=B1-4	3	-2

Используя формулы дешифрования, рассчитайте значения в ячейках I1:I5.

	G	H	I
1	=ОКРУГЛВНИЗ(СРЗНАЧА(A1:E1);0)	=ЕСЛИ(B1;МАКС(\$A\$2:\$C\$4);МИН(\$A\$2:\$C\$4))	=СУММ(G1:H1)
2	=ОКРУГЛВНИЗ(СРЗНАЧА(A2:E2);0)	=ЕСЛИ(B2;МАКС(\$A\$2:\$C\$4);МИН(\$A\$2:\$C\$4))	=СУММ(G2:H2)
3	=ОКРУГЛ(СРЗНАЧА(A3:E3);0)	=ЕСЛИ(B3;МАКС(\$A\$2:\$C\$4);МИН(\$A\$2:\$C\$4))	=СУММ(G3:H3)
4	=ОКРУГЛВВЕРХ(СРЗНАЧА(A4:E4);0)	=ЕСЛИ(B4;МАКС(\$A\$2:\$C\$4);МИН(\$A\$2:\$C\$4))	=СУММ(G4:H4)
5	=ОКРУГЛВВЕРХ(СРЗНАЧА(A5:E5);0)	=ЕСЛИ(B5;МАКС(\$A\$2:\$C\$4);МИН(\$A\$2:\$C\$4))	=СУММ(G5:H5)

По полученным значениям в ячейках I1:I5 определите, в соответствии с таблицей ниже, названия ячеек заминированных квадратов.

A1	A2	A3	A4	A5	B1	B2	B3	B4	B5	C1	C2	C3
-10	-9	-8	-7	-6	-5	-4	-3	-2	-1	0	1	2
C4	C5	D1	D2	D3	D4	D5	E1	E2	E3	E4	E5	
3	4	5	6	7	8	9	10	11	12	13	14	

В ответе названия ячеек запишите последовательно, слева на право, начиная со значения, соответствующего ячейке I1, затем I2, I3 и т.д.

Решение:

Для решения данной задачи необходимо рассчитать значения в ячейках G1:H5, затем найти значение сумм в колонке I1:I5.



	G	H	I	J
1	0	-8	-8	
2	-2	2	0	
3	-1	-8	-9	
4	-1	2	1	
5	1	2	3	

На основании значений в колонке J1:10 получаем названия требуемых ячеек.

Ответ:

A3	C1	A2	C2	C4
----	----	----	----	----

Задача 2 (10 баллов)

Честный человек и Лгун пытаются вспомнить день рождения своего друга (число дня и номер месяца рождения). Честный человек всегда говорит правду. Лгун всегда врёт

Честный человек:

- 1) Это зимний месяц.
- 2) В числе дня есть цифра 5.

Лгун:

- 1) Число дня двузначное.
- 2) Номер месяца меньше шести.

Когда день рождения друга Честного человека и Лгуна. Ответ поясните.

Решение:

Поскольку Честный человек всегда говорит правду, а Лгун всегда врёт мы объединим два высказывания по поводу номера месяца.

- 1) Честный человек: Это зимний месяц.
- 2) Лгун: Номер месяца меньше шести.

Из утверждения Честного человека понимаем, что номер месяца может быть 12, 1 и 2. А из утверждения Лгуна делаем вывод, что номер месяца больше или равен 6. Значит искомым месяцем – 12 (декабрь). Далее, из высказывания Лгуна «Число дня двузначное», понимаем, что число дня однозначное. А поскольку Честный человек сказал, что в числе дня есть цифра 5, делаем вывод, что число дня = 5. Получаем день рождения друга – 5.12 или 5 декабря.

Задача (10 баллов)

Аня придумала пароль в виде даты своего рождения, но в двоичной системе. Мысль об этом появилась у нее, когда на информатике в результирующем столбце логического



выражения она получила свои день и месяц рождения. Укажите дату рождения Ани, с помощью решения логических выражений и перевода результирующего столбца в десятичную систему счисления.

1. $A \rightarrow B \& C \leftrightarrow \bar{A}$
2. $A \& B \rightarrow C \leftrightarrow A$
3. $A \vee \bar{C} \rightarrow B \leftrightarrow A$

Решение:

Для того, чтобы узнать дату рождения Ани, необходимо найти результирующие столбцы логических выражений.

Для первой таблицы истинности:

A	B	C	\bar{A}	$B \& C$	$A \rightarrow B \& C$	$A \rightarrow B \& C \leftrightarrow \bar{A}$
0	0	0	1	0	1	1
0	0	1	1	0	1	1
0	1	0	1	0	1	1
0	1	1	1	1	1	1
1	0	0	0	0	0	1
1	0	1	0	0	0	1
1	1	0	0	0	0	1
1	1	1	0	1	1	0

Для перевода результата в дату необходимо разбить его на части, соответствующие числу и месяцу (каждые 2 двоичные цифры – одно десятичное число). Получается:

11 - 3

11 - 3

11 - 3

10 - 2

3332 – не является датой, соответственно, результат не подходящий.

Для второй таблицы истинности:

A	B	C	$A \& B$	$A \& B \rightarrow C$	$A \& B \rightarrow C \leftrightarrow A$
0	0	0	0	1	0
0	0	1	0	1	0
0	1	0	0	1	0



0	1	1	0	1	0
1	0	0	0	1	1
1	0	1	0	1	1
1	1	0	1	0	0
1	1	1	1	1	1

Для перевода результата в дату необходимо разбить его на части, соответствующие числу и месяцу (каждые 2 двоичные цифры – одно десятичное число). Получается:

00 - 0

00 - 0

11 - 3

01 - 1

0031 – не является датой, соответственно, выражение не подходит.

Для третьей таблицы истинности:

A	B	C	\bar{C}	$A \vee \bar{C}$	$A \vee \bar{C} \rightarrow B$	$A \vee \bar{C} \rightarrow B \leftrightarrow A$
0	0	0	1	1	0	1
0	0	1	0	0	1	0
0	1	0	1	1	1	0
0	1	1	0	0	1	0
1	0	0	1	1	0	0
1	0	1	0	1	0	0
1	1	0	1	1	1	1
1	1	1	0	1	1	1

Для перевода результата в дату необходимо разбить его на части, соответствующие числу и месяцу (каждые 2 двоичные цифры – одно десятичное число). Получается:

10 - 2

00 - 0

00 - 0

11 - 3

2003 – является датой 20.03.

Задача 4.

Катя знает, что IPv4-адрес его компьютера 192.168.0.107. Маска префикс /20. Помогите ей определить адрес сети, к какому классу относится данный ip-адрес, количество свободных узлов и порядковый номер компьютера.



Решение:

Маска префикс /20 является маской 255.255.240.0.

Чтобы определить адрес сети необходимо перевести IP-v4 адрес 192.168.0.107 и маску 255.255.255.0 в двоичную систему счисления.

$192.168.0.107 = 11000000.10101000.00000000.01101011$

$255.255.255.0 = 11111111.11111111.11110000.00000000$

Далее необходимо найти результат конъюнкции двоичных значений. В ответе получится 11000000.10101000.00000000.00000000. Переведя в десятичную систему 192.168.0.0 – адрес сети.

Зная, что в текущей подсети диапазон 0-20 можно определить количество широковещательных адресов по формулам:

- 1). $32-20=12$ – разность бит в адресе и крайнего значения диапазона;
- 2). $2^8=4096$ – определение количества подсетей.

Свободные узлы – это все узлы – 2, т.к. один из адресов зарезервирован под широковещательный адрес, а другой под адрес сети. Соответственно: $2^8-2=4094$.

IP-адрес относится к классу C, т.к. входит в диапазон 192.0.1.0-223.255.255.0.

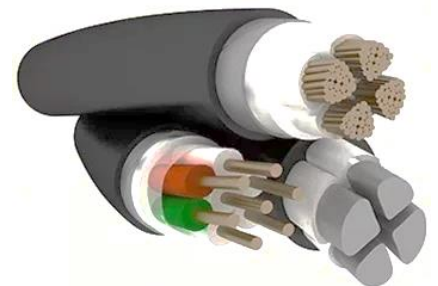
Для определения порядкового номера компьютера необходимо знать следующее – маску по префиксу, она будет выглядеть следующим образом: 255.255.240.0, затем перевести в двоичную систему $00_{10}=0000000_2$ и $107_{10}=01101011_2$. Нулевым битам маски соответствуют последние 4 бита числа 107, т.е. номер компьютера будет равен $1011_2=11_{10}$.

Ответ: 192.168.0.0 – адрес сети, IP-адрес 192.168.0.107 относится к классу C, количество свободных узлов 4094, номер компьютера 11.

Задача 5 (25 баллов)

ПРОВОДА

Коле было скучно, и он пошел к отцу в гараж, чтобы посмотреть, чем он там занимается. Среди множества различных инструментов он увидел много различных электрических кабелей. Его заинтересовало, что они имеют различную геометрическую форму, но всегда были обернуты в резиновую оболочку круглой формы.



Поскольку он увлекался геометрией, ему стало интересно проверить есть лучший способ для расположения проводов разной формы. Для проверки своей идеи он



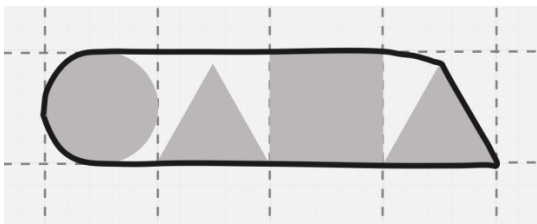
аккуратно расположил N геометрических фигур в прямоугольной сетке, разделенной на единичные квадраты, так чтобы в каждом квадрате была только одна геометрическая фигура.

Он решил размещать только три типа фигур:

Square – квадрат, стороны которого совпадают с квадратами, размеченной сетки.

Circle – круг, вписанный в единичный квадрат сетки.

Triangle – равносторонний треугольник, в котором одна сторона соответствует нижней стороне квадрата единичной сетки.



После этого он обернул все фигуры ниткой и измерять её, периодически изменяя положение и количество фигур в единичной сетке.

Помогите Коле и напишите программу, которая будет определять длину нитки, охватывающей различные фигуры, любого количества и расположения относительно друг друга, расположенных в прямоугольной единичной сетке.

Входные данные: в первой строке вам дается число (количество проводов), а во второй строка, состоящая из букв **S**, **C**, **T**, которые обозначают расположения слева направо квадрата, круга и треугольника соответственно.

Результат должен содержать вещественное число – длину нити, охватывающей все эти фигуры.

Несмотря на то, что в вашем задании указаны достаточно малые значения, ваша программа должна выполняться и для больших входных значений.

Пример:

Ввод: 4 CTST	Результат: 9.09
Ввод: 3 SCT	Результат: 7.51



Исходные данные:

15

SCTTCSTSSTTTTTT

Решение:

Функция **dist** вычисляет расстояние между двумя точками (x_1, y_1) и (x_2, y_2) в двумерном пространстве с помощью формулы расстояния между двумя точками в декартовой системе координат.

Функция **adjust** выполняет ряд вычислений в зависимости от входной строки **s** и переменной **n**. Она использует цикл `while` для поиска первого элемента строки **s**, отличного от 'T'. Затем она проверяет несколько условий и выполняет соответствующие вычисления в зависимости от значения `s[i]`.

Реализация на C++

```
#include <algorithm>
```

```
#include <cmath>
```

```
#include <cstdio>
```

```
#include <iostream>
```

```
#include <string>
```

```
using namespace std;
```

```
const double pi = acos(-1.0);
```

```
int n;
```

```
string s;
```

```
double dist(double x1, double y1, double x2, double y2) {
```

```
    double dx = x2 - x1;
```

```
    double dy = y2 - y1;
```

```
    return sqrt(dx * dx + dy * dy);
```

```
}
```

```
double adjust() {
```

```
    int i = 0;
```



```
while (i < n && s[i] == 'T') ++i;
if (i == n) return -0.5;
if (i == 0) {
    if (s[i] == 'S') return 0;
    else return -2 + pi / 2;
}
double x1 = 0.5;
double y1 = sqrt(3) / 2;
if (s[i] == 'S') {
    double x2 = i;
    double y2 = 1.0;
    return -x2 - 1 + dist(0, 0, x1, y1) + dist(x1, y1, x2, y2);
}
else {
    double xc = i + 0.5;
    double yc = 0.5;
    double h = dist(x1, y1, xc, yc);
    double alpha = atan2(y1 - yc, x1 - xc);
    double fi = acos(0.5 / h);
    double x2 = xc + 0.5 * cos(alpha - fi);
    double y2 = yc + 0.5 * sin(alpha - fi);
    return -xc - 1 + dist(0, 0, x1, y1) + dist(x1, y1, x2, y2) +
        0.5 * (alpha - fi - pi / 2);
}
}

int main() {
    cin >> n >> s;

    double ret = 2 * n + 2;
    ret += adjust();
    reverse(s.begin(), s.end());
    ret += adjust();
}
```




```
printf("%.2lf\n", ret);

return 0;
}
```

Реализация на Python

```
import math

n = int(input())
s = input()

def dist(x1, y1, x2, y2):
    dx = x2 - x1
    dy = y2 - y1
    return math.sqrt(dx * dx + dy * dy)

def adjust():
    i = 0
    while i < n and s[i] == 'T':
        i += 1
    if i == n:
        return -0.5
    if i == 0:
        if s[i] == 'S':
            return 0
        else:
            return -2 + math.pi / 2
    x1 = 0.5
    y1 = math.sqrt(3) / 2
    if s[i] == 'S':
        x2 = i
```



```
y2 = 1.0
return -x2 - 1 + dist(0, 0, x1, y1) + dist(x1, y1, x2, y2)
else:
    xc = i + 0.5
    yc = 0.5
    h = dist(x1, y1, xc, yc)
    alpha = math.atan2(y1 - yc, x1 - xc)
    fi = math.acos(0.5 / h)
    x2 = xc + 0.5 * math.cos(alpha - fi)
    y2 = yc + 0.5 * math.sin(alpha - fi)
    return -xc - 1 + dist(0, 0, x1, y1) + dist(x1, y1, x2, y2) + 0.5 * (alpha - fi - math.pi / 2)

ret = 2 * n + 2
ret += adjust()
s = s[::-1]
ret += adjust()
print("{:.2f}".format(ret))
```

Демонстрация решение на C++



```
Granit2024.cpp  X
Granit2024      (Глобальная область)  main()
29
30     double x1 = 0.5;
31     double y1 = sqrt(3) / 2;
32     if (s[i] == 'S') {
33         double x2 = i;
34         double y2 = 1.0;
35         return -x2 - 1 + dist(0, 0, x1, y1) + dist(x1, y1, x2, y2);
36     }
37     else {
38         double xc = i + 0.5;
39         double yc = 0.5;
40         double h = dist(x1, y1, xc, yc);
41         double alpha = atan2(y1 - yc, x1 - xc);
42         double fi = acos(0.5 / h);
43         double x2 = xc + 0.5 * cos(alpha - fi);
44         double y2 = yc + 0.5 * sin(alpha - fi);
45         return -xc - 1 + dist(0, 0, x1, y1) + dist(x1, y1, x2, y2) +
46             0.5 * (alpha - fi - pi / 2);
47     }
48 }
49
50 int main() {
51     cin >> n >> s;
52
53     double ret = 2 * n + 2;
54     ret += adjust();
55     reverse(s.begin(), s.end());
56     ret += adjust();
57     printf("%.2lf\n", ret);
58
59     return 0;
60 }
```

```
Консоль отладки Microsoft V  X
15
SC TTCSTSSSTTTTTT
31.50
```

Ответ:

31.50

Задача 6 (25 баллов)

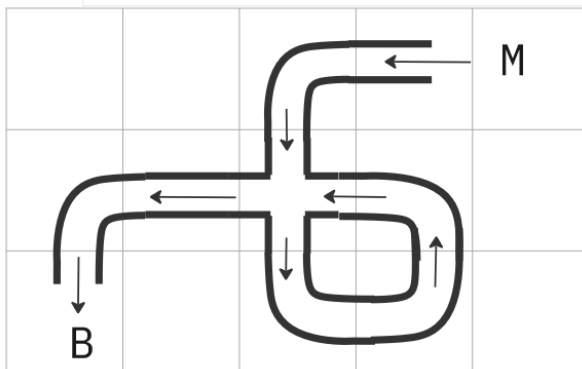
СИЛА СИБИРИ

Компания Газпром заключила контракт с компанией в Китае и решила построить новый газопровод. Чтобы помочь в проектировании нового газопровода, который будет использоваться для доставки российского газа в Китай, Пекин и Москва используют компьютерную программу GasPipe. В программе страна разделена на строки R и столбцы C . Каждая ячейка может быть пустой или содержать один из семи основных строительных блоков газопровода:



I	-	+	1	2	3	4

Газ идет от Чаяндинского месторождения во Владивосток. Газ может течь в любом направлении через секции трубы. Блок «+» особенный тем, что газ должен течь в двух направлениях (вертикальном и горизонтальном), как в следующий пример:



Работа над новым газопроводом уже началась, когда было обнаружено, что злоумышленники завладели планом и удалили из него ровно один строительный блок, то есть заменили его пустой ячейкой.

Напишите программу, определяющую, откуда был удален блок и какого он типа.

Входные данные:

В первой строке записаны два целых числа R и C — размеры страны ($1 \leq R, C \leq 25$). Следующие R строк содержат план, каждая из которых состоит ровно из C символов.

Типы блоков:

- Точка ('.'), обозначающая пустую ячейку;
- Символы 'I', '-', '+', '1', '2', '3', '4', представляющие типы строительных блоков газопровода;
- Буквы «М» и «В» обозначают Чаяндинское месторождение и Владивосток. Каждый из них появится ровно один раз в плане.

Расход газа будет однозначно определен на входе; ровно по одному квадрату будет соседствовать с месторождением и Владивостоком. Кроме того, в плане не будет избыточных блоков, т. е. все блоки в плане должны быть использованы после



Олимпиада школьников «Гранит науки» - 2024

добавления недостающего блока. Входные данные будут такими, что решение будет существовать и будет уникальным.

Результат должен содержать строку и столбец удаленного блока, а также тип блока (один из семи символов как на входе).

Несмотря на то, что в вашем задании указаны достаточно малые значения, ваша программа должна выполняться и для больших входных значений.

Пример:

Ввод: 3 7M-.-V.	Результат: 2 4 -
Ввод: 3 5 ..1-M 1-+.. V.23.	Результат: 2 4 4

Исходные данные:

8 10
.....
.M.14.....
.I1+3.....
.23I.14...
...I...I...
...2-+3...
.....2-V..
.....

Решение:



Сначала напишем четыре вспомогательные функции `ToLeft(r,c)`, `toRight(r,c)`, `toUp(r,c)` и `toDown(r,c)`, сообщающие нам, может ли газ течь в в каждом направлении из пустой ячейки `(r,c)`. Например, функция `ToLeft(r,c)` может выглядеть следующим образом:

```
return true if Map(r,s-1) = '+' or Map(r,s-1) = '-' or Map(r,s-1) = '1' or Map(r,s-1) = '2'
```

Задача может быть решена за один проход по карте. Для каждой пустой ячейки `(r, c)` мы выполняем следующие проверки:

```
if toLeft(r,c) and toRight(r,c) and toUp(r,c) and toDown(r,c) then output r c '+'
```

```
else if toLeft(r,c) and toRight(r,c) then output r c '-'
```

```
else if toUp(r,c) and toDown(r,c) then output r c '|'
```

```
else if toRight(r,c) and toDown(r,c) then output r c '1'
```

```
else if toRight(r,c) and toUp(r,c) then output r c '2'
```

```
else if toLeft(r,c) and toUp(r,c) then output r c '3'
```

```
else if toLeft(r,c) and toDown(r,c) then output r c '4'
```

Реализация на C++

```
#define _CRT_SECURE_NO_WARNINGS
```

```
#include <cstdio>
```

```
#define MAXR 25
```

```
#define MAXS 25
```

```
int R, S;
```

```
char a[MAXR][MAXS + 1];
```

```
int toLeft(int r, int s) {
```

```
    return s > 0 && (a[r][s - 1] == '+' || a[r][s - 1] == '-' || a[r][s - 1] == '1' || a[r][s - 1] == '2');
```

```
}
```

```
int toRight(int r, int s) {
```

```
    return s < S - 1 && (a[r][s + 1] == '+' || a[r][s + 1] == '-' || a[r][s + 1] == '3' || a[r][s + 1] == '4');
```

```
}
```



```
int toUp(int r, int s) {
    return r > 0 && (a[r - 1][s] == '+' || a[r - 1][s] == 'I' || a[r - 1][s] == '1' || a[r - 1][s] == '4');
}

int toDown(int r, int s) {
    return r < R - 1 && (a[r + 1][s] == '+' || a[r + 1][s] == 'I' || a[r + 1][s] == '2' || a[r + 1][s] ==
'3');
}

int main(void) {
    scanf("%d%d", &R, &S);
    for (int r = 0; r < R; ++r) scanf("%s", a[r]);

    for (int r = 0; r < R; ++r)
        for (int s = 0; s < S; ++s) {
            if (a[r][s] != '.') continue;

            if (toLeft(r, s) && toRight(r, s) && toUp(r, s) && toDown(r, s)) printf("%d %d +\n", r +
1, s + 1);
            else if (toLeft(r, s) && toRight(r, s)) printf("%d %d -\n", r + 1, s + 1);
            else if (toUp(r, s) && toDown(r, s)) printf("%d %d I\n", r + 1, s + 1);
            else if (toRight(r, s) && toDown(r, s)) printf("%d %d 1\n", r + 1, s + 1);
            else if (toRight(r, s) && toUp(r, s)) printf("%d %d 2\n", r + 1, s + 1);
            else if (toLeft(r, s) && toUp(r, s)) printf("%d %d 3\n", r + 1, s + 1);
            else if (toLeft(r, s) && toDown(r, s)) printf("%d %d 4\n", r + 1, s + 1);

        }
    return 0;
}
```

Реализация на Python

MAXR = 25

MAXS = 25



```
def to_left(r, s):
    return s > 0 and (a[r][s-1]== '+' or a[r][s-1]== '-' or a[r][s-1]== '1' or a[r][s-1]== '2')

def to_right(r, s):
    return s < S-1 and (a[r][s+1]== '+' or a[r][s+1]== '-' or a[r][s+1]== '3' or a[r][s+1]== '4')

def to_up(r, s):
    return r > 0 and (a[r-1][s]== '+' or a[r-1][s]== 'I' or a[r-1][s]== '1' or a[r-1][s]== '4')

def to_down(r, s):
    return r < R-1 and (a[r+1][s]== '+' or a[r+1][s]== 'I' or a[r+1][s]== '2' or a[r+1][s]== '3')

R, S = map(int, input().split())
a = [input() for _ in range(R)]

for r in range(R):
    for s in range(S):
        if a[r][s] != '.':
            continue

        if to_left(r,s) and to_right(r,s) and to_up(r,s) and to_down(r,s):
            print(f"{r+1} {s+1} +")
        elif to_left(r,s) and to_right(r,s):
            print(f"{r+1} {s+1} -")
        elif to_up(r,s) and to_down(r,s):
            print(f"{r+1} {s+1} I")
        elif to_right(r,s) and to_down(r,s):
            print(f"{r+1} {s+1} 1")
        elif to_right(r,s) and to_up(r,s):
            print(f"{r+1} {s+1} 2")
        elif to_left(r,s) and to_up(r,s):
            print(f"{r+1} {s+1} 3")
```




elif to_left(r,s) and to_down(r,s):

```
print(f"{r+1} {s+1} 4")
```

Демонстрация

The screenshot shows a C++ program in a Visual Studio Code editor. The code defines three functions: toUp, toDown, and main. The main function reads a grid of characters and prints the coordinates of the first cell that is not a dot and has neighbors in all four directions (up, down, left, right).

```
16 int toUp(int r, int s) {
17     return r > 0 && (a[r - 1][s] == '+' || a[r - 1][s] == '|' || a[r - 1][s] == '1' || a[r - 1][s] == '4');
18 }
19 int toDown(int r, int s) {
20     return r < R - 1 && (a[r + 1][s] == '+' || a[r + 1][s] == '|' || a[r + 1][s] == '2' || a[r + 1][s] == '3');
21 }
22
23 int main(void) {
24     scanf("%d%d", &R, &S);
25     for (int r = 0; r < R; ++r) scanf("%s", a[r]);
26
27     for (int r = 0; r < R; ++r)
28         for (int s = 0; s < S; ++s) {
29             if (a[r][s] != '.') continue;
30
31             if (toLeft(r, s) && toRight(r, s) && toUp(r, s) && toDown(r, s)) printf("%d %d +\n", r + 1, s + 1);
32             else if (toLeft(r, s) && toRight(r, s)) printf("%d %d -\n", r + 1, s + 1);
33             else if (toUp(r, s) && toDown(r, s)) printf("%d %d |\n", r + 1, s + 1);
34             else if (toRight(r, s) && toDown(r, s)) printf("%d %d 1\n", r + 1, s + 1);
35             else if (toRight(r, s) && toUp(r, s)) printf("%d %d 2\n", r + 1, s + 1);
36             else if (toLeft(r, s) && toUp(r, s)) printf("%d %d 3\n", r + 1, s + 1);
37             else if (toLeft(r, s) && toDown(r, s)) printf("%d %d 4\n", r + 1, s + 1);
38         }
39     return 0;
40 }
41
42
```

The output window shows the following text:

```
8 10
.....
.M.14.....
.I1+3.....
.23I.14...
Вывод: ...I..I...
Показать: ...2+3...
.....2-B..
.....
5 6 |
```

Ответ:

5 6 I



ИНФОРМАТИКА

ВАРИАНТ 4

Задача 1 (10 баллов)

Перед вами зашифрованная карта минных полей.

	A	B	C	D	E
1		=2>-5	-5	A1	3
2	3	-1	-5	-4	E4+2
3	3	0	6	=C2+7	D2
4	5	0	-4	0	=D3<C1
5	4	1	3	3	-2

Используя формулы дешифрования, рассчитайте значения в ячейках I1:I5.

	G	H	I
1	=ОКРУГЛВНИЗ(СРЗНАЧА(A1:E1);0)	=ЕСЛИ(B1;МАКС(\$A\$2:\$C\$4);МИН(\$A\$2:\$C\$4))	=СУММ(G1:H1)
2	=ОКРУГЛВНИЗ(СРЗНАЧА(A2:E2);0)	=ЕСЛИ(B2;МАКС(\$A\$2:\$C\$4);МИН(\$A\$2:\$C\$4))	=СУММ(G2:H2)
3	=ОКРУГЛ(СРЗНАЧА(A3:E3);0)	=ЕСЛИ(B3;МАКС(\$A\$2:\$C\$4);МИН(\$A\$2:\$C\$4))	=СУММ(G3:H3)
4	=ОКРУГЛВВЕРХ(СРЗНАЧА(A4:E4);0)	=ЕСЛИ(B4;МАКС(\$A\$2:\$C\$4);МИН(\$A\$2:\$C\$4))	=СУММ(G4:H4)
5	=ОКРУГЛВВЕРХ(СРЗНАЧА(A5:E5);0)	=ЕСЛИ(B5;МАКС(\$A\$2:\$C\$4);МИН(\$A\$2:\$C\$4))	=СУММ(G5:H5)

По полученным значениям в ячейках I1:I5 определите, в соответствии с таблицей ниже, названия ячеек заминированных квадратов.

A1	A2	A3	A4	A5	B1	B2	B3	B4	B5	C1	C2	C3
-10	-9	-8	-7	-6	-5	-4	-3	-2	-1	0	1	2
C4	C5	D1	D2	D3	D4	D5	E1	E2	E3	E4	E5	
3	4	5	6	7	8	9	10	11	12	13	14	

В ответе названия ячеек запишите последовательно, слева на право, начиная со значения, соответствующего ячейке I1, затем I2, I3 и т.д.

Решение:

Для решения данной задачи необходимо рассчитать значения в ячейках G1:H5, затем найти значение сумм в колонке I1:I5.



	G	H	I	J
1	0	6	6	
2	-1	6	5	
3	2	-5	-3	
4	1	-5	-4	
5	2	6	8	

На основании значений в колонке J1:10 получаем названия требуемых ячеек.

Ответ:

D2	D1	B3	B2	D4
----	----	----	----	----

Задача 2 (10 баллов)

Честный человек и Лгун пытаются вспомнить день рождения своего друга (число дня и номер месяца рождения). Честный человек всегда говорит правду. Лгун всегда врёт

Честный человек:

- 1) Это весенний месяц.
- 2) В числе дня есть цифра 7.

Лгун:

- 1) Число дня двузначное.
- 2) Номер месяца меньше или равен четырём.

Когда день рождения друга Честного человека и Лгуна. Ответ поясните.

Решение:

Поскольку Честный человек всегда говорит правду, а Лгун всегда врёт мы объединим два высказывания по поводу номера месяца.

- 1) Честный человек: Это весенний месяц.
- 2) Номер месяца меньше или равен четырём.

Из утверждения Честного человека понимаем, что номер месяца может быть 3, 4 и 5. А из утверждения Лгуна делаем вывод, что номер месяца больше 4. Значит искомым месяц – 5 (май). Далее, из высказывания Лгуна «Число дня двузначное», понимаем, что число дня однозначное. А поскольку Честный человек сказал, что в числе дня есть цифра 7, делаем вывод, что число дня = 7. Получаем день рождения друга – 7.05 или 7 мая.

Задача 3 (10 баллов)

Аня придумала пароль в виде даты своего рождения, но в двоичной системе. Мысль об этом появилась у нее, когда на информатике в результирующем столбце логического



выражения она получила дату рождения. Укажите дату рождения Ани, решив логические выражения и переведя их в десятичную систему счисления.

1. $A \rightarrow \bar{B} \leftrightarrow C \wedge A$

2. $\bar{A} \vee B \rightarrow C \wedge B$

3. $A \wedge C \rightarrow B \leftrightarrow C$

Решение:

Для того, чтобы узнать дату рождения Ани, необходимо найти результирующие столбцы логических выражений.

Для первой таблицы истинности:

A	B	C	\bar{B}	$C \wedge A$	$A \rightarrow B$	$A \rightarrow \bar{B} \leftrightarrow C \wedge A$
0	0	0	1	0	1	0
0	0	1	1	0	1	0
0	1	0	0	0	1	0
0	1	1	0	0	1	0
1	0	0	1	0	1	0
1	0	1	1	1	1	1
1	1	0	0	0	0	1
1	1	1	0	1	0	0

Для перевода результата в дату необходимо разбить его на части, соответствующие числу и месяцу (каждые 2 двоичные цифры – одно десятичное число). Получается:

00 - 0

00 - 0

01 - 1

10 - 2

0012 – не является датой, соответственно, этот ответ не подходит.

Для второй таблицы истинности:

A	B	C	\bar{A}	$C \wedge B$	$\bar{A} \vee B$	$\bar{A} \vee B \rightarrow C \wedge B$
0	0	0	1	0	1	0
0	0	1	1	0	1	0
0	1	0	1	0	1	0
0	1	1	1	1	1	1
1	0	0	0	0	0	1



Олимпиада школьников «Гранит науки» - 2024

1	0	1	0	0	0	1
1	1	0	0	0	1	0
1	1	1	0	1	1	1

Для перевода результата в дату необходимо разбить его на части, соответствующие числу и месяцу (каждые 2 двоичные цифры – одно десятичное число). Получается:

00 - 0

01 - 1

11 - 3

01 - 1

0131 – не является датой, соответственно, результат не подходящий.

0031 – не является датой, соответственно, выражение не подходит.

Для третьей таблицы истинности:

A	B	C	$A \wedge C$	$A \wedge C \rightarrow B$	$A \wedge C \rightarrow B \leftrightarrow C$
0	0	0	0	1	0
0	0	1	0	1	1
0	1	0	0	1	0
0	1	1	0	1	1
1	0	0	0	1	0
1	0	1	1	0	0
1	1	0	0	1	0
1	1	1	1	1	1

Для перевода результата в дату необходимо разбить его на части, соответствующие числу и месяцу (каждые 2 двоичные цифры – одно десятичное число). Получается:

01 - 1

01 - 1

00 - 0

01 - 1

1101 – является датой 11.01.

Ответ. 11.01

Задача 4 (20 баллов)

Катя знает, что IPv4-адрес его компьютера 192.168.56.12. Маска префикс /28. Помогите ей определить адрес сети, к какому классу относится данный ip-адрес, количество свободных узлов и порядковый номер компьютера.



Решение:

Маска префикс /24 является маской 255.255.255.240.

Чтобы определить адрес сети необходимо перевести IP-v4 адрес 192.168.56.1 и маску 255.255.255.0 в двоичную систему счисления.

$192.168.56.1 = 11000000.10101000.00111000.11110000$

$255.255.255.0 = 11111111.11111111.11111111.00000000$

Далее необходимо найти результат конъюнкции двоичных значений. В ответе получится $11000000.10101000.00111000.00000000$. Переведя в десятичную систему $192.168.56.0$ – адрес сети.

Зная, что в текущей подсети диапазон 0-28 можно определить количество широковещательных адресов по формулам:

- 1). $32-28=4$ – разность бит в адресе и крайнего значения диапазона;
- 2). $2^4=16$ – определение количества подсетей.

Свободные узлы – это все узлы – 2, т.к. один из адресов зарезервирован под широковещательный адрес, а другой под адрес сети. Соответственно: $2^4-2=14$.

IP-адрес относится к классу C, т.к. входит в диапазон 192.0.1.0-223.255.255.0.

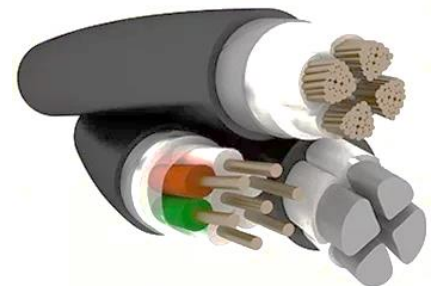
Для определения порядкового номера компьютера необходимо знать следующее – маску по префиксу, она будет выглядеть следующим образом: 255.255.255.240, затем перевести в двоичную систему $240_{10} = 11110000_2$ и $12_{10} = 00001100_2$. Нулевым битам маски соответствуют последние 4 бита числа 107, т.е. номер компьютера будет равен $1100_2 = 12_{10}$.

Ответ: 192.168.56.0 – адрес сети, IP-адрес 192.168.56.12 относится к классу C, количество свободных узлов 14, номер компьютера 12.

Задача 5 (25 баллов)

ПРОВОДА

Коле было скучно, и он пошел к отцу в гараж, чтобы посмотреть, чем он там занимается. Среди множества различных инструментов он увидел много различных электрических кабелей. Его заинтересовало, что они имеют различную геометрическую форму, но всегда были обернуты в резиновую оболочку круглой формы.



Поскольку он увлекался геометрией, ему стало интересно проверить есть лучший способ для расположения проводов разной формы. Для проверки своей идеи он



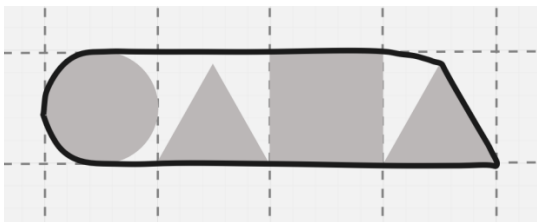
аккуратно расположил N геометрических фигур в прямоугольной сетке, разделенной на единичные квадраты, так чтобы в каждом квадрате была только одна геометрическая фигура.

Он решил размещать только три типа фигур:

Square – квадрат, стороны которого совпадают с квадратами, размеченной сетки.

Circle – круг, вписанный в единичный квадрат сетки.

Triangle – равносторонний треугольник, в котором одна сторона соответствует нижней стороне квадрата единичной сетки.



После этого он обернул все фигуры ниткой и измерять её, периодически изменяя положение и количество фигур в единичной сетке.

Помогите Коле и напишите программу, которая будет определять длину нитки, охватывающей различные фигуры, любого количества и расположения относительно друг друга, расположенных в прямоугольной единичной сетке.

Входные данные: в первой строке вам дается число (количество проводов), а во второй строка, состоящая из букв **S**, **C**, **T**, которые обозначают расположения слева направо квадрата, круга и треугольника соответственно.

Результат должен содержать вещественное число – длину нити, охватывающей все эти фигуры.

Несмотря на то, что в вашем задании указаны достаточно малые значения, ваша программа должна выполняться и для больших входных значений.

Пример:

Ввод: 4 CTST	Результат: 9.09
Ввод: 3 SCT	Результат: 7.51



Исходные данные:

19

CSCTSTSTTCSTSSCTCSS

Решение:

Функция **dist** вычисляет расстояние между двумя точками (x_1, y_1) и (x_2, y_2) в двумерном пространстве с помощью формулы расстояния между двумя точками в декартовой системе координат.

Функция **adjust** выполняет ряд вычислений в зависимости от входной строки **s** и переменной **n**. Она использует цикл `while` для поиска первого элемента строки **s**, отличного от 'T'. Затем она проверяет несколько условий и выполняет соответствующие вычисления в зависимости от значения $s[i]$.

Реализация на C++

```
#include <algorithm>
```

```
#include <cmath>
```

```
#include <cstdio>
```

```
#include <iostream>
```

```
#include <string>
```

```
using namespace std;
```

```
const double pi = acos(-1.0);
```

```
int n;
```

```
string s;
```

```
double dist(double x1, double y1, double x2, double y2) {
```

```
    double dx = x2 - x1;
```

```
    double dy = y2 - y1;
```

```
    return sqrt(dx * dx + dy * dy);
```

```
}
```

```
double adjust() {
```

```
    int i = 0;
```




```
while (i < n && s[i] == 'T') ++i;
if (i == n) return -0.5;
if (i == 0) {
    if (s[i] == 'S') return 0;
    else return -2 + pi / 2;
}
double x1 = 0.5;
double y1 = sqrt(3) / 2;
if (s[i] == 'S') {
    double x2 = i;
    double y2 = 1.0;
    return -x2 - 1 + dist(0, 0, x1, y1) + dist(x1, y1, x2, y2);
}
else {
    double xc = i + 0.5;
    double yc = 0.5;
    double h = dist(x1, y1, xc, yc);
    double alpha = atan2(y1 - yc, x1 - xc);
    double fi = acos(0.5 / h);
    double x2 = xc + 0.5 * cos(alpha - fi);
    double y2 = yc + 0.5 * sin(alpha - fi);
    return -xc - 1 + dist(0, 0, x1, y1) + dist(x1, y1, x2, y2) +
        0.5 * (alpha - fi - pi / 2);
}
}

int main() {
    cin >> n >> s;

    double ret = 2 * n + 2;
    ret += adjust();
    reverse(s.begin(), s.end());
    ret += adjust();
}
```



```
printf("%.2lf\n", ret);

return 0;
}
```

Реализация на Python

```
import math

n = int(input())
s = input()

def dist(x1, y1, x2, y2):
    dx = x2 - x1
    dy = y2 - y1
    return math.sqrt(dx * dx + dy * dy)

def adjust():
    i = 0
    while i < n and s[i] == 'T':
        i += 1
    if i == n:
        return -0.5
    if i == 0:
        if s[i] == 'S':
            return 0
        else:
            return -2 + math.pi / 2
    x1 = 0.5
    y1 = math.sqrt(3) / 2
    if s[i] == 'S':
        x2 = i
```



```
y2 = 1.0
```

```
return -x2 - 1 + dist(0, 0, x1, y1) + dist(x1, y1, x2, y2)
```

```
else:
```

```
xc = i + 0.5
```

```
yc = 0.5
```

```
h = dist(x1, y1, xc, yc)
```

```
alpha = math.atan2(y1 - yc, x1 - xc)
```

```
fi = math.acos(0.5 / h)
```

```
x2 = xc + 0.5 * math.cos(alpha - fi)
```

```
y2 = yc + 0.5 * math.sin(alpha - fi)
```

```
return -xc - 1 + dist(0, 0, x1, y1) + dist(x1, y1, x2, y2) + 0.5 * (alpha - fi - math.pi / 2)
```

```
ret = 2 * n + 2
```

```
ret += adjust()
```

```
s = s[::-1]
```

```
ret += adjust()
```

```
print("{:.2f}".format(ret))
```

Демонстрация решение на C++



```
Granit2024.cpp  X
Granit2024      (Глобальная область)  adjust()
29
30     double x1 = 0.5;
31     double y1 = sqrt(3) / 2;
32     if (s[i] == 'S') {
33         double x2 = i;
34         double y2 = 1.0;
35         return -x2 - 1 + dist(0, 0, x1, y1) + dist(x1, y1, x2, y2);
36     }
37     else {
38         double xc = i + 0.5;
39         double yc = 0.5;
40         double h = dist(x1, y1, xc, yc);
41         double alpha = atan2(y1 - yc, x1 - xc);
42         double fi = acos(0.5 / h);
43         double x2 = xc + 0.5 * cos(alpha - fi);
44         double y2 = yc + 0.5 * sin(alpha - fi);
45         return -xc - 1 + dist(0, 0, x1, y1) + dist(x1, y1, x2, y2) +
46             0.5 * (alpha - fi - pi / 2);
47     }
48 }
49
50 int main() {
51     cin >> n >> s;
52
53     double ret = 2 * n + 2;
54     ret += adjust();
55     reverse(s.begin(), s.end());
56     ret += adjust();
57     printf("%.2lf\n", ret);
58
59     return 0;
60 }
```

```
Консоль отладки Microsoft V  X
19
CSCTSTSTTCSTSSCTCSS
39.57
```

Ответ:

39.57

Задача 6.

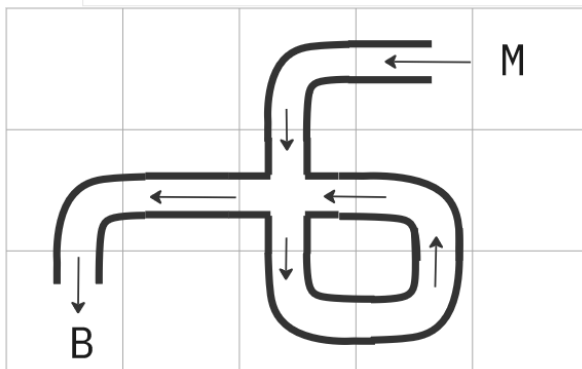
СИЛА СИБИРИ

Компания Газпром заключила контракт с компанией в Китае и решила построить новый газопровод. Чтобы помочь в проектировании нового газопровода, который будет использоваться для доставки российского газа в Китай, Пекин и Москва используют компьютерную программу GasPipe. В программе страна разделена на строки R и столбцы C . Каждая ячейка может быть пустой или содержать один из семи основных строительных блоков газопровода:



I	-	+	1	2	3	4

Газ идет от Чаяндинского месторождения во Владивосток. Газ может течь в любом направлении через секции трубы. Блок «+» особенный тем, что газ должен течь в двух направлениях (вертикальном и горизонтальном), как в следующий пример:



Работа над новым газопроводом уже началась, когда было обнаружено, что злоумышленники завладели планом и удалили из него ровно один строительный блок, то есть заменили его пустой ячейкой.

Напишите программу, определяющую, откуда был удален блок и какого он типа.

Входные данные:

В первой строке записаны два целых числа R и C — размеры страны ($1 \leq R, C \leq 25$). Следующие R строк содержат план, каждая из которых состоит ровно из C символов.

Типы блоков:

- Точка ('.'), обозначающая пустую ячейку;
- Символы 'I', '-', '+', '1', '2', '3', '4', представляющие типы строительных блоков газопровода;
- Буквы «М» и «В» обозначают Чаяндинское месторождение и Владивосток. Каждый из них появится ровно один раз в плане.

Расход газа будет однозначно определен на входе; ровно по одному квадрату будет соседствовать с месторождением и Владивостоком. Кроме того, в плане не будет избыточных блоков, т. е. все блоки в плане должны быть использованы после



Олимпиада школьников «Гранит науки» - 2024

добавления недостающего блока. Входные данные будут такими, что решение будет существовать и будет уникальным.

Результат должен содержать строку и столбец удаленного блока, а также тип блока (один из семи символов как на входе).

Несмотря на то, что в вашем задании указаны достаточно малые значения, ваша программа должна выполняться и для больших входных значений.

Пример:

Ввод: 3 7М-.-В.	Результат: 2 4 -
Ввод: 3 5 ..1-М 1-+.. В.23.	Результат: 2 4 4

Исходные данные:

8 10
.....
.М..4.....
.I1+3.....
.23I.14...
...I.II...
...2-+3...
.....2-Z..
.....

Решение:



Сначала напишем четыре вспомогательные функции `ToLeft(r,c)`, `toRight(r,c)`, `toUp(r,c)` и `toDown(r,c)`, сообщающие нам, может ли газ течь в в каждом направлении из пустой ячейки `(r,c)`. Например, функция `ToLeft(r,c)` может выглядеть следующим образом:

```
return true if Map(r,s-1) = '+' or Map(r,s-1) = '-' or Map(r,s-1) = '1' or Map(r,s-1) = '2'
```

Задача может быть решена за один проход по карте. Для каждой пустой ячейки `(r, c)` мы выполняем следующие проверки:

```
if toLeft(r,c) and toRight(r,c) and toUp(r,c) and toDown(r,c) then output r c '+'  
else if toLeft(r,c) and toRight(r,c) then output r c '-'  
else if toUp(r,c) and toDown(r,c) then output r c '|'  
else if toRight(r,c) and toDown(r,c) then output r c '1'  
else if toRight(r,c) and toUp(r,c) then output r c '2'  
else if toLeft(r,c) and toUp(r,c) then output r c '3'  
else if toLeft(r,c) and toDown(r,c) then output r c '4'
```

Реализация на C++

```
#define _CRT_SECURE_NO_WARNINGS  
  
#include <cstdio>  
  
#define MAXR 25  
#define MAXS 25  
  
int R, S;  
char a[MAXR][MAXS + 1];  
int toLeft(int r, int s) {  
    return s > 0 && (a[r][s - 1] == '+' || a[r][s - 1] == '-' || a[r][s - 1] == '1' || a[r][s - 1] == '2');  
}  
int toRight(int r, int s) {
```



```
return s < S - 1 && (a[r][s + 1] == '+' || a[r][s + 1] == '-' || a[r][s + 1] == '3' || a[r][s + 1] ==
'4');
}
int toUp(int r, int s) {
return r > 0 && (a[r - 1][s] == '+' || a[r - 1][s] == '1' || a[r - 1][s] == '1' || a[r - 1][s] == '4');
}
int toDown(int r, int s) {
return r < R - 1 && (a[r + 1][s] == '+' || a[r + 1][s] == '1' || a[r + 1][s] == '2' || a[r + 1][s] ==
'3');
}

int main(void) {
scanf("%d%d", &R, &S);
for (int r = 0; r < R; ++r) scanf("%s", a[r]);

for (int r = 0; r < R; ++r)
for (int s = 0; s < S; ++s) {
if (a[r][s] != '.') continue;

if (toLeft(r, s) && toRight(r, s) && toUp(r, s) && toDown(r, s)) printf("%d %d +\n", r +
1, s + 1);
else if (toLeft(r, s) && toRight(r, s)) printf("%d %d -\n", r + 1, s + 1);
else if (toUp(r, s) && toDown(r, s)) printf("%d %d I\n", r + 1, s + 1);
else if (toRight(r, s) && toDown(r, s)) printf("%d %d 1\n", r + 1, s + 1);
else if (toRight(r, s) && toUp(r, s)) printf("%d %d 2\n", r + 1, s + 1);
else if (toLeft(r, s) && toUp(r, s)) printf("%d %d 3\n", r + 1, s + 1);
else if (toLeft(r, s) && toDown(r, s)) printf("%d %d 4\n", r + 1, s + 1);

}
return 0;
}
```




Реализация на Python

MAXR = 25

MAXS = 25

```
def to_left(r, s):
```

```
    return s > 0 and (a[r][s-1]== '+' or a[r][s-1]== '-' or a[r][s-1]== '1' or a[r][s-1]== '2')
```

```
def to_right(r, s):
```

```
    return s < S-1 and (a[r][s+1]== '+' or a[r][s+1]== '-' or a[r][s+1]== '3' or a[r][s+1]== '4')
```

```
def to_up(r, s):
```

```
    return r > 0 and (a[r-1][s]== '+' or a[r-1][s]== 'I' or a[r-1][s]== '1' or a[r-1][s]== '4')
```

```
def to_down(r, s):
```

```
    return r < R-1 and (a[r+1][s]== '+' or a[r+1][s]== 'I' or a[r+1][s]== '2' or a[r+1][s]== '3')
```

```
R, S = map(int, input().split())
```

```
a = [input() for _ in range(R)]
```

```
for r in range(R):
```

```
    for s in range(S):
```

```
        if a[r][s] != '.':
```

```
            continue
```

```
            if to_left(r,s) and to_right(r,s) and to_up(r,s) and to_down(r,s):
```

```
                print(f"{r+1} {s+1} +")
```

```
            elif to_left(r,s) and to_right(r,s):
```

```
                print(f"{r+1} {s+1} -")
```

```
            elif to_up(r,s) and to_down(r,s):
```

```
                print(f"{r+1} {s+1} I")
```

```
            elif to_right(r,s) and to_down(r,s):
```

```
                print(f"{r+1} {s+1} 1")
```

```
            elif to_right(r,s) and to_up(r,s):
```



```
print(f"{r+1} {s+1} 2")
elif to_left(r,s) and to_up(r,s):
    print(f"{r+1} {s+1} 3")
elif to_left(r,s) and to_down(r,s):
    print(f"{r+1} {s+1} 4")
```

Демонстрация

The screenshot shows a C++ IDE with a file named Granit2024.cpp. The code defines three helper functions: toUp, toDown, and toLeft. The main function reads a grid size R and S, then iterates through the grid. For each cell (r, s), it checks the four directions (up, down, left, right) and prints the first valid move. The output shows a sequence of moves: 8 10, 11+3, 231.14, 2-Z, and 2 4 1.

```
16 int toUp(int r, int s) {
17     return r > 0 && (a[r - 1][s] == '+' || a[r - 1][s] == '|' || a[r - 1][s] == '1' || a[r - 1][s] ==
18 }
19 int toDown(int r, int s) {
20     return r < R - 1 && (a[r + 1][s] == '+' || a[r + 1][s] == '|' || a[r + 1][s] == '2' || a[r + 1][s]
21 }
22
23 int main(void) {
24     scanf("%d%d", &R, &S);
25     for (int r = 0; r < R; ++r) scanf("%s", a[r]);
26
27     for (int r = 0; r < R; ++r)
28         for (int s = 0; s < S; ++s) {
29             if (a[r][s] != '.') continue;
30
31             if (toLeft(r, s) && toRight(r, s) && toUp(r, s) && toDown(r, s)) printf("%d %d +\n", r +
32 else if (toLeft(r, s) && toRight(r, s)) printf("%d %d -\n", r + 1, s + 1);
33 else if (toUp(r, s) && toDown(r, s)) printf("%d %d |\n", r + 1, s + 1);
34 else if (toRight(r, s) && toDown(r, s)) printf("%d %d 1\n", r + 1, s + 1);
35 else if (toRight(r, s) && toUp(r, s)) printf("%d %d 2\n", r + 1, s + 1);
36 else if (toLeft(r, s) && toUp(r, s)) printf("%d %d 3\n", r + 1, s + 1);
37 else if (toLeft(r, s) && toDown(r, s)) printf("%d %d 4\n", r + 1, s + 1);
38
39         }
40     return 0;
41 }
42
```

Консоль отладки Microsoft V

```
8 10
.....
.M..4.....
.I1+3.....
Вер: 23I.14...
Поиск: I.II. данные из: Сборка
.....2-Z..
.....
2 4 1
```

Ответ:

2 4 1