

Финансовый инжиниринг

2022/23 учебный год

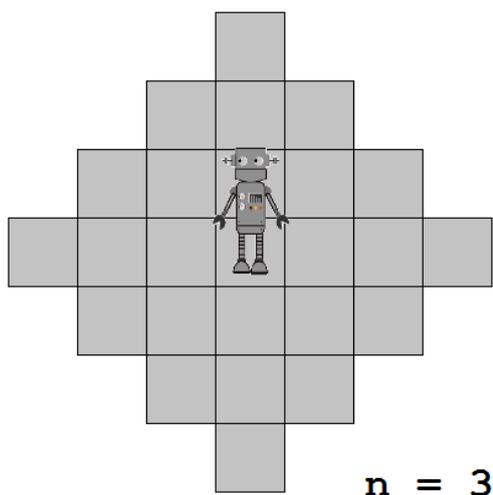
Заключительный этап

Предметный тур

Информатика. 8–11 класс

Задача VI.1.1.1. Прогулка робота (10 баллов)

Алевтина тестирует разработанного её командой робота. Этот робот может перемещаться по клетчатой поверхности, с одной клетки на соседнюю с ней по стороне. В начальный момент робот стоит в некоторой клетке поля. Заряда его аккумулятора хватит на n шагов. Алевтине интересно, а в сколько клетках может оказаться робот при этих условиях. Для примера, предположим, что робот может сделать не более трех шагов. Тогда множество всех достижимых им клеток представлено на следующем рисунке и содержит 25 клеток.



По заданному n нужно определить количество клеток, в которых может оказаться робот, сделав не более n шагов.

Формат входных данных

На вход подается одно целое число n — количество шагов, на которое у робота хватит энергии. $0 \leq n \leq 10^9$.

Формат выходных данных

Вывести количество клеток, в которых может оказаться робот, сделав не более n шагов.

Примеры

Пример №1

Стандартный ввод
3
Стандартный вывод
25

Пример №2

Стандартный ввод
0
Стандартный вывод
1

Пример программы-решения

Ниже представлено решение на языке C++.

```
1  #include <bits/stdc++.h>
2  using namespace std;
3
4  signed main(){
5      int n;
6      cin >> n;
7      int ans = n * n;
8      n++;
9      ans += n * n;
10     cout << ans;
11 }
```

Ниже представлено решение на языке Python 3.

```
1  n = int(input())
2  print(n * n + (n + 1) * (n + 1))
```

Задача VI.1.1.2. Музыкальные модули (15 баллов)

Михаил изобрел принципиально новый музыкальный инструмент. Инструмент состоит из нескольких модулей. Количество модулей произвольно и может меняться путем добавления каких-то из них к инструменту или же их удаления из его состава. Один модуль так же является не самым красивым по звучанию, но инструментом.

Готовый инструмент характеризуется объемом и тоном. Объем задается натуральным числом от 1 до n , а тон это целое число от $-n$ до n . Чем меньше отношение абсолютного значения тона к размеру объема (это отношение назовем звучанием), тем красивее звучит инструмент. Но если тон становится точно равен 0, инструмент теряет звучание. При равенстве вышеуказанных отношений, красивее звучит тот инструмент, у которого больше объем.

Главным достижением Михаила при разработке нового инструмента является то, что он сумел реализовать механизм объединения модулей с взаимным гашением тонов разного знака. При таком объединении, объем полученного инструмента равен сумме объемов исходных модулей, а тон так же равен сумме исходных тонов, но взятых со своим знаком. **Абсолютное значение тона любого исходного модуля всегда равно его объему.** Значение тона у любого модуля всегда целое число, не равное 0.

Для примера, пусть есть два модуля с тонами -9 и 2 . Тогда их можно объединить и получить инструмент с тоном -7 и объемом 11 .

На данный момент Михаил сумел построить звучащий инструмент с тоном a и объемом b . Теперь он хочет создать еще один модуль так, чтобы добавление этого модуля к уже построенному инструменту давало бы самое красивое возможное звучание. Помогите Михаилу выбрать значение тона для дополнительного модуля.

Формат входных данных

На вход подается два целых числа a и b через пробел — значения тона и объема уже построенного Михаилом инструмента. $-1000 \leq a \leq 1000, a \neq 0. 1 \leq b \leq 1000. |a| \leq b$.

Формат выходных данных

В первую строку вывести значение тона для дополнительного модуля, добавление которого сделает звучание наиболее красивым.

Во вторую строку вывести два числа через пробел — значение тона и объема для самого красиво звучащего инструмента, которого можно добиться добавлением ровно одного модуля к уже построенному Михаилом инструменту.

Примеры

Пример №1

Стандартный ввод
10 30
Стандартный вывод
-11
-1 41

Пример №2

Стандартный ввод
-1 3
Стандартный вывод
2
1 5

Пример программы-решения

Ниже представлено решение на языке C++.

```
1  #include <bits/stdc++.h>
2  #define int long long
3  using namespace std;
4
5  signed main(){
6      int a, b;
7      cin >> a >> b;
8      double bnum = 1, bdenum = 1, m, snum;
9
10     for(int i = -10000; i <= 10000; i++){
11         double num = abs(a + i);
12         if(num == 0)
13             continue;
14         double denum = abs(i) + b;
15
16         if(num / denum < bnum / bdenum || abs(num / denum - bnum / bdenum) < 1e-7
17             ↪ && denum > bdenum){
18             bnum = num;
19             snum = a + i;
20             bdenum = denum;
21             m = i;
22         }
23     }
24     cout.precision(0);
25     cout <<fixed << m<<"\n"<< snum<<' '<<bdenum;
26 }
```

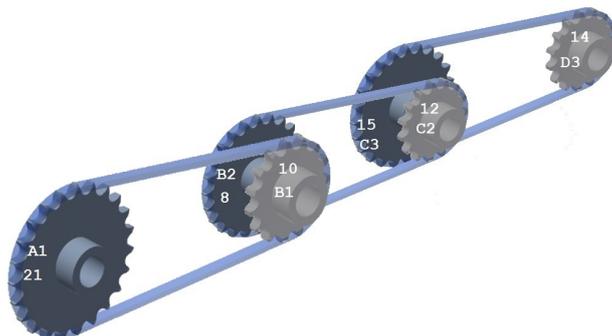
Ниже представлено решение на языке Python 3.

```
1  a, b = map(int, input().split())
2  mini = 10 ** 18
3  ans = -1
4  for i in range(-20000, 20001):
5      if i != -a and abs((a + i) / (b + abs(i))) < mini:
6          mini = abs((a + i) / (b + abs(i)))
7          ans = i
8  print(ans)
9  print(a + ans, b + abs(ans))
```

Задача VI.1.1.3. Цепная передача (20 баллов)

Алексею на день рождения родители купили набор «Юный механик». Помимо прочих очень интересных деталей, в набор входят несколько шестеренок с различным количеством зубцов и передаточных цепей к ним. Их можно соединять на специальном стенде произвольным образом в очень сложные системы для передачи вращающего момента. Алексей тут же собрал линейную цепочку из нескольких попарно соединенных шестерен. При этом самая левая и самая правая шестерни в цепочке были на своих отдельных вращающихся валах, а промежуточные шестерни от разных цепей парами были помещены на общие валы. Более точно это можно описать следующим образом: например, Алексей соединил общими цепями шестерни A_1 и B_1 , шестерни B_2 и C_2 , а так же шестерни C_3 и D_3 . Тогда на отдельный вращающийся

вал будут помещены: шестерня A_1 , шестерня D_3 , одновременно две шестерни B_1 и B_2 , а так же одновременно две шестерни C_2 и C_3 . Для лучшего понимания смотрите рисунок.



Если вращать шестерню A_1 , то будут вращаться и все остальные. Число оборотов будет определяться соотношением числа зубцов у смежных шестерен. Например, если две смежные шестерни A_1 с 21 зубцом и B_1 с 10 зубцами соединены цепью, то полный оборот шестерни A_1 будет соответствовать 2.1 оборота шестерни B_1 . Для того, чтобы обе шестерни сделали целое число оборотов, нужно A_1 вращать 10 раз, тогда B_1 провернется 21 раз.

На рисунке, помимо обозначения шестерен, указано количество зубцов у каждой из них (для данного примера). Теперь Алексей хочет понять, какое наименьшее **целое** количество оборотов шестерни A_1 нужно сделать, чтобы последняя в цепочке шестерня так же сделала **целое** количество оборотов.

Продолжим рассмотрения примера на рисунке. Утверждается, что если сделать 2 оборота шестерней A_1 , то шестерня D_3 сделает ровно 3 оборота. Действительно, 2 оборота шестерни A_1 вызовут 4.2 оборота шестерни B_1 . Такое же количество оборотов сделает и шестерня B_2 с 8 зубцами. Тогда шестерня C_2 с 12 зубцами сделает 2.8 оборота. Соответственно шестерня C_3 с 15 зубцами также сделает 2.8 оборота, а значит шестерня D_3 с 14 зубцами сделает 3 оборота.

Вам нужно решить эту задачу для произвольного (не очень большого) числа пар шестерен с заданными количествами зубцов у каждой.

Формат входных данных

В первой строке содержится количество пар шестерен n , соединенных цепью. Шестерни соединены в линейную цепочку аналогично примеру на рисунке. $1 \leq n \leq 5$.

В следующих n строках содержатся через пробел количества зубцов в каждой паре соединенных цепью шестерен. Вторая шестерня предыдущей пары и первая шестерня следующей пары помещаются на общий отдельный вращающийся вал. Первая шестерня первой пары и вторая шестерня последней пары находятся каждая на своем отдельном вале. Число зубцов в каждой шестерне находится в пределах от 3 до 100.

Формат выходных данных

Вывести два числа через пробел — наименьшее количество целых оборотов первой шестерни первой пары, и соответствующее ему число целых оборотов второй шестерни последней пары.

Примеры

Пример №1

Стандартный ввод
3 21 10 8 12 15 14
Стандартный вывод
2 3

Пример №2

Стандартный ввод
1 21 10
Стандартный вывод
10 21

Пример №3

Стандартный ввод
5 3 3 4 4 5 5 6 6 7 7
Стандартный вывод
1 1

Пример программы-решения

Ниже представлено решение на языке C++.

```
1  #include <bits/stdc++.h>
2  #define sz(a) (int)a.size()
3  #define pb push_back
4  #define all(a) a.begin(), a.end()
5  #define for0(i, n) for(int i = 0; i < n; i++)
6  #define int long long
7  using namespace std;
8
9  signed main(){
10     int n;
11     cin >> n;
12     int num = 1, denum = 1;
13     for0(i, n){
14         int a, b;
15         cin >> a >> b;
```

```
16     num *= a;
17     denum *= b;
18 }
19 int g = __gcd(num, denum);
20 num /= g;
21 denum /= g;
22 cout << denum << ' ' << num << endl;
23 }
```

Ниже представлено решение на языке Python 3.

```
1 import math
2 n = int(input())
3 num1 = 1
4 num2 = 1
5 for i in range(n):
6     a, b = map(int, input().split())
7     num1 *= a
8     num2 *= b
9 gcd = math.gcd(num1, num2)
10 num2 //= gcd
11 num1 //= gcd
12 print(num2, num1)
```

Задача VI.1.1.4. Футбольный контроль (25 баллов)

Фёдор начинающий футбольный болельщик. Он узнал, что раз в четыре года проходит чемпионат мира по футболу. Чемпионат делится на две стадии - групповую и плей-офф. На групповой стадии все участвующие команды случайно делятся на группы по четыре команды. Далее внутри каждой группы все команды играют между собой. Таким образом, каждая команда на групповой стадии играет три матча, а внутри группы происходит шесть игр. В итоге для каждой группы получается турнирная таблица, в которой эти четыре команды выстраиваются от самой сильной до самой слабой. Две самых сильных команды выходят в следующую стадию - плей-офф и далее проводятся игры на выбывание.

Для построения таблицы используются следующие показатели (в порядке приоритетности, если все предыдущие показатели у двух команд равны, сравниваются следующие по списку).

1. Количество набранных командой очков. За победу начисляется три очка, за ничью одно, за поражение очки не начисляют. Очки за все сыгранные командой матчи суммируются.

2. При равенстве очков у двух команд, выше в таблице стоит та, у которой больше разница забитых и пропущенных мячей по итогам всех трех сыгранных этой командой игр.

3. При равенстве разницы забитых и пропущенных мячей у двух команд, выше в таблице стоит та, у которой больше забитых мячей.

В списке было еще много условий, вплоть до подсчёта желтых карточек и жеребьевки, но Фёдор далее решил не читать и ограничиться приведёнными выше правилами. Теперь он хочет проверить правильность построения турнирной таблицы. Фёдор знает результаты всех шести матчей в группе, осталось понять, в каком порядке команды будут следовать в таблице.

Рассмотрим пример. Допустим, в одной группе играют команды Armenia, Belarus, Costa Rica и Denmark. Далее приведены результаты игр между ними:

Armenia 3:1 Belarus

Belarus 3:1 Costa Rica

Costa Rica 4:3 Armenia

Armenia 1:1 Denmark

Denmark 0:3 Belarus

Costa Rica 1:5 Denmark

По итогам игр получается, что

- Belarus две игры выиграла, одну проиграла и набрала 6 очков;
- Armenia одну игру выиграла, одну проиграла, одну сыграла вничью и набрала 4 очка;
- Denmark одну игру выиграла, одну проиграла, одну сыграла вничью и набрала 4 очка;
- Costa Rica одну игру выиграла, две проиграла и набрала 3 очка.

Пока известно, что Belarus заняла первое место в таблице и вышла в плей-офф, но за второе выходящее место спорят команды Armenia и Denmark.

Armenia забила 7 мячей и пропустила 6 мячей. Разница равна +1.

Denmark забила 6 мячей и пропустила 5 мячей. Разница равна +1.

По второму показателю у этих команд снова равенство.

Armenia забила больше, чем Denmark, поэтому в таблице команда Armenia будет на втором месте, а Denmark на третьем.

Costa Rica займет четвертую позицию.

В итоге в плей-офф выйдут Belarus и Armenia.

Формат входных данных

На вход в шести строках подаются данные об итогах шести матчей между четырьмя командами в формате, указанном в примере. Названия команд состоят из больших и малых английских букв и знаков одиночного пробела и могут состоять из нескольких слов. Число символов в названии команд не превосходит 50. Команда от количества забитых голов отделена одним пробелом, сам счет разделён двоеточием без пробела. Количество забитых мячей в матче не превосходит 99. Все команды в группе имеют попарно различные названия.

Формат выходных данных

В первую строку вывести заголовок «Standings:»

Далее в следующие четыре строки вывести команды в том порядке, в каком они будут следовать в таблице. Если для двух или более команд порядок по итогам указанных условий не определен, вывести их в алфавитном порядке. Для каждой команды указать перед её именем в таблице через пробел место, которое она заняла в группе. Если несколько команд имеют полностью одинаковые показатели, перед

каждой из этих команд вывести интервал мест, в которые она попадает. Интервал выводить в виде «первое место интервала - последнее место интервала» и далее пробел перед именем соответствующей команды. Для лучшего понимания смотрите примеры.

Далее вывести пустую строку.

Далее в следующей строке вывести сообщение «Advance to the playoffs:»

Далее, если вышедшие команды определены, в две строки вывести две вышедшие из группы в плей-офф команды. Если среди этих двух команд есть лидер – в первую из этих строк вывести его, иначе, если обе вышедшие команды полностью равны – вывести их в алфавитном порядке. Если по итогам турнира нельзя однозначно определить две вышедшие команды, вывести одну строку с сообщением «Lottery».

Примеры

Пример №1

Стандартный ввод
Armenia 3:1 Belarus Belarus 3:1 Costa Rica Costa Rica 4:3 Armenia Armenia 1:1 Denmark Denmark 0:3 Belarus Costa Rica 1:5 Denmark
Стандартный вывод
Standings: 1 Belarus 2 Armenia 3 Denmark 4 Costa Rica Advance to the playoffs: Belarus Armenia

Пример №2

Стандартный ввод
Armenia 3:3 Belarus Armenia 0:2 Costa Rica Armenia 0:1 Denmark Belarus 0:1 Costa Rica Belarus 0:2 Denmark Costa Rica 1:1 Denmark
Стандартный вывод
Standings: 1-2 Costa Rica 1-2 Denmark 3-4 Armenia 3-4 Belarus Advance to the playoffs: Costa Rica Denmark

Пример №3

Стандартный ввод
Armenia 0:3 Belarus Armenia 0:3 Costa Rica Armenia 0:3 Denmark Belarus 2:2 Costa Rica Belarus 2:2 Denmark Costa Rica 2:2 Denmark
Стандартный вывод
Standings: 1-3 Belarus 1-3 Costa Rica 1-3 Denmark 4 Armenia Advance to the playoffs: Lottery

Пример программы-решения

Ниже представлено решение на языке C++.

```
1 #include <bits/stdc++.h>
2 #define sz(a) (int)a.size()
3 #define pb push_back
4 #define all(a) a.begin(), a.end()
5 #define for0(i, n) for(int i = 0; i < n; i++)
6 #define for1(i, n) for(int i = 1; i <= n; i++)
7
8 using namespace std;
9 typedef pair<int, int> pii;
```

```

10 typedef vector<int> vi;
11 typedef vector<vector<int> > vvi;
12 typedef map<string, int> msi;
13
14 int lss(string n1, string n2, msi &ts, msi &tm, msi &p){
15
16     if(p[n1] < p[n2])
17         return 1;
18
19     int d1 = ts[n1] - tm[n1];
20     int d2 = ts[n2] - tm[n2];
21
22     if(p[n1] == p[n2] && d1 < d2)
23         return 1;
24
25     if(p[n1] == p[n2] && d1 == d2 && ts[n1] < ts[n2])
26         return 1;
27
28     if(p[n1] == p[n2] && d1 == d2 && ts[n1] == ts[n2])
29         return 0;
30
31     return -1;
32 }
33
34
35 signed main(){
36
37     msi total_scored, total_missed, points;
38     set<string> S;
39     string s;
40     for0(i, 6){
41         getline(cin, s);
42         int t1 = 0;
43         while(s[t1] < '0' || s[t1] > '9')
44             t1++;
45         t1--;
46
47         int t2 = sz(s) - 1;
48         while(s[t2] < '0' || s[t2] > '9')
49             t2--;
50         t2++;
51
52         string n1 = s.substr(0, t1);
53         string n2 = s.substr(t2 + 1);
54         string score = s.substr(t1 + 1, t2 - t1 - 1);
55
56         stringstream ss;
57         ss << score;
58         int a1, a2;
59         char c;
60         ss >> a1 >> c >> a2;
61
62         total_scored[n1] += a1;
63         total_scored[n2] += a2;
64         total_missed[n1] += a2;
65         total_missed[n2] += a1;
66
67         if(a1 == a2){
68             points[n1]++;
69             points[n2]++;

```

```

70     }
71     if(a1 < a2){
72         points[n2] += 3;
73     }
74     if(a1 > a2){
75         points[n1] += 3;
76     }
77     S.insert(n1);
78     S.insert(n2);
79 }
80 vector<string> tab;
81 for(auto q : S)
82     tab.pb(q);
83 for(int i = 0; i < 3; i++)
84     for(int j = i + 1; j < 4; j++){
85         if(lss(tab[i], tab[j], total_scored, total_missed, points) == 1)
86             swap(tab[i], tab[j]);
87         if(lss(tab[i], tab[j], total_scored, total_missed, points) == 0 &&
88             ↪ tab[i] > tab[j])
89             swap(tab[i], tab[j]);
90     }
91 map<string, int> fp, lp;
92 int t = 1, dt = 0;
93 fp[tab[0]] = t;
94 lp[tab[0]] = t;
95 for(int i = 1; i < sz(tab); i++){
96     if(lss(tab[i], tab[i - 1], total_scored, total_missed, points) == 1){
97         dt = 0;
98         t++;
99         fp[tab[i]] = t;
100     }
101     else
102         if(lss(tab[i], tab[i - 1], total_scored, total_missed, points) == 0){
103             dt++;
104             t++;
105             fp[tab[i]] = fp[tab[i - 1]];
106         }
107     for(int j = 0; j < dt + 1; j++){
108         lp[tab[i - j]] = fp[tab[i - j]] + dt;
109     }
110 }
111 cout<<"Standings:"<< endl;
112 for(auto q : tab){
113     cout<< fp[q];
114     if(fp[q] != lp[q])
115         cout << "-"<<lp[q];
116     cout << ' ' << q << endl;
117 }
118 cout << endl<<"Advance to the playoffs:"<< endl;
119 if(lss(tab[2], tab[1], total_scored, total_missed, points) == 1){
120     for0(i, 2)
121         cout << tab[i]<< endl;
122 }
123 else
124     if(lss(tab[2], tab[1], total_scored, total_missed, points) == 0)
125         cout << "Lottery\n";

```

Ниже представлено решение на языке Java.

```
1 import java.util.*;
```

```

2 public class Main {
3     public static void add(Map<String, Integer> p, Integer val, String name) {
4         if (!p.containsKey(name))
5             p.put(name, val);
6         else
7             p.put(name, val + p.get(name));
8     }
9     public static class Info implements Comparable<Info>{
10        String name;
11        Integer scores;
12        Integer points;
13        Integer scoresDelta;
14
15        public Info(String name, Integer scores, Integer points, Integer
16        ↪ scoresDelta) {
17            this.name = name;
18            this.scores = scores;
19            this.points = points;
20            this.scoresDelta = scoresDelta;
21        }
22
23        @Override
24        public int compareTo(Info o) {
25            if (points > o.points)
26                return -1;
27            if (points < o.points)
28                return 1;
29            if (scoresDelta > o.scoresDelta)
30                return -1;
31            if (scoresDelta < o.scoresDelta)
32                return 1;
33            if (scores > o.scores)
34                return -1;
35            if (scores < o.scores)
36                return 1;
37            return -o.name.compareTo(name);
38        }
39        public int compare(Info o) {
40            if (points > o.points)
41                return -1;
42            if (points < o.points)
43                return 1;
44            if (scoresDelta > o.scoresDelta)
45                return -1;
46            if (scoresDelta < o.scoresDelta)
47                return 1;
48            return o.scores.compareTo(scores);
49        }
50    }
51    public static void compare(String name1, String name2) {
52    }
53    public static void main(String[] args) {
54        Map<String, Integer> scores = new HashMap<>();
55        Map<String, Integer> points = new HashMap<>();
56        Map<String, Integer> scoresDelta = new HashMap<>();
57        Scanner readLine = new Scanner(System.in);
58        for (int i = 0; i < 6; ++i) {
59            String line = readLine.nextLine();
60            Scanner sc = new Scanner(line);
61            String name = "";

```

```

61     String in = "123";
62     do {
63         in = sc.next();
64         if (!in.contains(":")) {
65             if (name.length() > 0)
66                 name += " ";
67             name += in;
68         }
69     }
70     while (!in.contains(":"));
71     String name1 = name;
72     String[] vals = in.split(":");
73     String name2 = "";
74     while (sc.hasNext()) {
75         if (name2.length() > 0)
76             name2 += " ";
77         name2 += sc.next();
78     }
79     Integer point1 = Integer.parseInt(vals[0]);
80     Integer point2 = Integer.parseInt(vals[1]);
81     add(scores, point1, name1);
82     add(scores, point2, name2);
83     if (point1 > point2) {
84         add(points, 3, name1);
85         add(points, 0, name2);
86     }
87     if (point1 < point2) {
88         add(points, 3, name2);
89         add(points, 0, name1);
90     }
91     if (point1.equals(point2)) {
92         add(points, 1, name1);
93         add(points, 1, name2);
94     }
95     add(scoresDelta, point1 - point2, name1);
96     add(scoresDelta, point2 - point1, name2);
97 }
98 List<Info> infos2 = new ArrayList<>();
99 for (Map.Entry<String, Integer> entry : points.entrySet()) {
100     String name = entry.getKey();
101     Integer point = entry.getValue();
102     Integer scoreDelta = scoresDelta.get(name);
103     Integer score = scores.get(name);
104     Info info = new Info(name, score, point, scoreDelta);
105     infos2.add(info);
106 }
107 Info[] infos = infos2.toArray(new Info[0]);
108 Arrays.sort(infos);
109 System.out.println("Standings:");
110 for (int i = 0; i < 4; ++i) {
111     int cnt = 0;
112     int mini = -1;
113     for (int j = 0; j < 4; ++j) {
114         if (infos[i].compare(infos[j]) == 0) {
115             cnt++;
116             if (mini == -1)
117                 mini = j;
118         }
119     }
120 }

```

```

121         int l = mini + 1;
122         int r = l + cnt - 1;
123         if (l == r)
124             System.out.print(l + " ");
125         else
126             System.out.print(l + "-" + r + " ");
127         System.out.println(infos[i].name);
128     }
129     System.out.println();
130     System.out.println("Advance to the playoffs:");
131     if (infos[0].compare(infos[2]) == -1 && infos[1].compare(infos[2]) == -1)
132     ↪ {
133         if (infos[0].compare(infos[1]) == -1) {
134             System.out.println(infos[0].name);
135             System.out.println(infos[1].name);
136         }
137         else {
138             if (infos[0].name.compareTo(infos[1].name) < 0){
139                 System.out.println(infos[0].name);
140                 System.out.println(infos[1].name);
141             }
142             else {
143                 System.out.println(infos[1].name);
144                 System.out.println(infos[0].name);
145             }
146         }
147     }
148     else
149         System.out.println("Lottery");
150 }

```

Задача VI.1.1.5. Цепочки проектов (30 баллов)

Эдуард Леонидович — директор исследовательского института. Каждый день его институт либо начинает ровно один новый проект либо заканчивает ровно один из уже начатых. Таким образом, если мы обозначим проекты их номерами и выпишем для каждого дня номер начинаемого или заканчиваемого проекта, то получим расписание работы института в этот период времени.

Например пусть последовательность работ над проектами выглядит следующим образом: 2, 4, 5, 5, 3, 2, 1, 4, 1, 3.

Она означает, что: в первый день институт начал проект номер 2, во второй день институт начал проект номер 4, в третий день институт начал проект номер 5, в четвертый день институт закончил проект номер 5, в пятый день институт начал проект номер 3, в шестой день институт закончил проект номер 2, в седьмой день институт начал проект номер 1, в восьмой день институт закончил проект номер 4, в девятый день институт закончил проект номер 1, в десятый день институт закончил проект номер 3.

В отчетности, которую заполняет Эдуард Леонидович, считается, что проект В является продолжением проекта А, если проект В начался между датами начала и окончания проекта А, и закончился позже окончания проекта А. Наибольшую ценность имеют цепочки проектов. Набор проектов A_1, A_2, \dots, A_k является цепочкой, если для любых двух соседних проектов A_i и A_{i+1} в этой цепочке верно, что проект A_{i+1} является продолжением проекта A_i .

В вышеприведенном примере можно выделить следующие цепочки проектов:

1, 2, 3, 4, 5 имеют длину один,

2–4, 2–3, 4–3, 4–1 имеют длину два,

2–4–1, 2–4–3 имеют длину три.

Помогите с отчетом Эдуарду Леонидовичу. По списку начатых и законченных институтом проектов определите длину самой большой цепочки проектов, выполненной институтом за отчетный период.

Формат входных данных

В первой строке задается количество проектов n , выполненных институтом в отчетный период. $1 \leq n \leq 10^5$.

Во второй строке задана последовательность выполнения проектов. Она состоит из $2 \cdot n$ чисел через пробел. Каждое из этих чисел от 1 до n и каждое встречается в последовательности ровно два раза. Первое вхождение соответствует началу проекта с этим номером, второе вхождение — окончанию.

Формат выходных данных

Вывести одно число — длину самой большой цепочки проектов за отчетный период.

Примеры

Пример №1

Стандартный ввод
5 2 4 5 5 3 2 1 4 1 3
Стандартный вывод
3

Пример программы-решения

Ниже представлено решение на языке C++.

```
1  #include <bits/stdc++.h>
2  #define sz(a) (int)a.size()
3  #define pb push_back
4  #define all(a) a.begin(), a.end()
5  #define for0(i, n) for(int i = 0; i < n; i++)
6  #define for1(i, n) for(int i = 1; i <= n; i++)
7  #define int long long
8
9  using namespace std;
10 typedef long long ll;
11 typedef pair<int, int> pii;
12 typedef vector<int> vi;
13
```

```

14 int LG = 19;
15 int N = (1 << LG);
16 vi t(2 * N, 0);
17
18 void upd(int a, int pos){
19     pos += N;
20     t[pos] = a;
21     pos /= 2;
22     while(pos){
23         t[pos] = max(t[2 * pos], t[2 * pos + 1]);
24         pos /= 2;
25     }
26 }
27
28 int getMax(int l, int r){
29     int res = -1;
30     l += N;
31     r += N;
32     while(l <= r){
33         if(l % 2 == 1){
34             res = max(res, t[l]);
35         }
36         if(r % 2 == 0){
37             res = max(res, t[r]);
38         }
39         l = (l + 1) / 2;
40         r = (r - 1) / 2;
41     }
42     return res;
43 }
44
45 signed main(){
46     int n;
47     cin >> n;
48     vi v(2 * n), next(n + 1, -1);
49     for0(i, 2 * n){
50         cin >> v[i];
51     }
52
53     for(int i = 2 * n - 1; i >= 0; i--){
54         if(next[v[i]] == -1)
55             next[v[i]] = i;
56     }
57     for0(i, 2 * n){
58         if(next[v[i]] != i){
59             int mx = getMax(i, next[v[i]]);
60             upd(mx + 1, next[v[i]]);
61         }
62     }
63     cout << getMax(0, 2 * n - 1);
64 }

```

Ниже представлено решение на языке Python 3.

```

1 # Python3 Code Addition
2 import math
3 # limit for array size
4 N = 200005;
5
6 # Max size of tree

```

```

7 tree = [0] * (2 * N);
8
9 # function to build the tree
10 def build(arr) :
11
12     # insert leaf nodes in tree
13     for i in range(n) :
14         tree[n + i] = arr[i];
15
16     # build the tree by calculating parents
17     for i in range(n - 1, 0, -1) :
18         tree[i] = tree[i << 1] + tree[i << 1 | 1];
19
20 # function to update a tree node
21 def update(p, value) :
22
23     # set value at position p
24     tree[p + n] = value;
25     p = p + n;
26
27     # move upward and update parents
28     i = p;
29
30     while i > 1 :
31
32         tree[i >> 1] = max(tree[i], tree[i ^ 1]);
33         i >>= 1;
34
35 # function to get sum on interval [l, r)
36 def query(l, r) :
37
38     res = 0;
39
40     # loop to find the sum in the range
41     l += n;
42     r += n;
43     maxi = 0
44     while l < r :
45
46         if (l & 1) :
47             maxi = max(maxi, tree[l])
48             l += 1
49
50         if (r & 1) :
51             r -= 1;
52             maxi = max(maxi, tree[r])
53
54         l >>= 1;
55         r >>= 1
56
57     return maxi;
58
59
60
61 n = int(input())
62 n *= 2
63 a = list(map(int, input().split()))
64 last = [0] * (n + 1)
65 sz = [0] * n
66 for i in range(n - 1, -1, -1):

```

```
67     if last[a[i]] == 0:
68         last[a[i]] = i
69 for i in range(n):
70     lst = last[a[i]]
71     if lst == i:
72         continue
73     l = i + 1
74     r = lst - 1
75     val = query(l, r + 1)
76     update(lst, val + 1)
77 print(query(0, 2 * n - 1 + 1))
```

Тестовые наборы для задач представлены по ссылке — <https://disk.yandex.ru/d/bImms4sGhSMn1Q>.

Математика. 8–9 классы

Задача VI.1.2.1. (10 баллов)

Темы: алгебра, текстовые задачи.

Теплоходы «Зосима Шашков» и «Мустай Карим» компании «Водоход» вышли из Москвы в круиз по Волге по маршруту Москва — Нижний Новгород — Москва. По пути они сделали остановку в Ярославле. При этом известно, что «Зосима Шашков» плыл вдвое больше времени, чем стоял в Ярославле «Мустай Карим», а «Мустай Карим» плыл втрое больше времени, чем стоял в Ярославле «Зосима Шашков». Дополнительно известно, что время круиза по маршруту Москва — Нижний Новгород — Москва одинаково для всех теплоходов, следующих по нему. На сколько процентов один из теплоходов плавает быстрее другого?

Решение

Пусть x — время в пути «Зосимы Шашков», y — время стоянки «Мустай Карим». Тогда можно составить следующую схему.

Зосима	Шашков	Мустай	Карим
Плыл	Стоял	Плыл	Стоял
$2x$	y	$3y$	x

Из того, что, время в круизе одинаково, получаем: $2x + y = 3y + x \Rightarrow 2y = x$. Следовательно, время в пути «Мустай Карим» — $3x/2$. Что составляет 75% от $2x$. Поэтому ответ: на 25%.

Ответ: 25.

Критерии оценивания

1. Обосновано получен верный ответ — 10 баллов.
2. Верно построена математическая модель — 5 баллов.
3. Решение не соответствует ни одному из критериев, перечисленных выше — 0 баллов.

Задача VI.1.2.2. (20 баллов)

Темы: алгебра, преобразования, уравнения.

Известно, что Олег Игнатьевич, Кирилл Евгеньевич и Анастасия Валерьевна готовы вложиться в стартап «Умники» в количестве x^2 , y^2 и z^2 соответственно. При этом суммы вложений связаны соотношением: $x^2 + 4y^2 + 9z^2 = 9$. Функция дохода стартапа имеет вид: $8x - 8y + 3z$. Оценить возможные потери вкладчиков при реализации наихудшего сценария.

Решение

Введем два вектора так, чтобы выражение $8x - 8y + 3z$ представляло их скалярное произведение. Например, $\bar{a} = (x, 2y, 3z)$ и $\bar{b} = (8, -4, 1)$. Поскольку, из определения

скалярного произведения следует, что

$$|\bar{a} \cdot \bar{b}| \leq |\bar{a}| \cdot |\bar{b}| \Rightarrow |8x - 8y + 3z| \leq 3 \cdot 9 = 27.$$

Поэтому, значения, которые может принимать выражение $8x - 8y + 3z$ есть отрезок $[-27; 27]$. И наименьшее значение равно -27 .

Ответ: -27.

Критерии оценивания

1. Обосновано получен верный ответ — 20 баллов.
2. Получены неверные ответы из-за вычислительной ошибки, но при этом имеется верная последовательность всех шагов решения — 15 баллов.
3. Есть существенные ошибки, но прослеживается логика решения — 10 баллов.
4. Решение не соответствует ни одному из критериев, перечисленных выше — 0 баллов.

Задача VI.1.2.3. (30 баллов)

Темы: алгебра, геометрическая прогрессия.

Результат инвестирования Александра Максимовича Денежкина в проект «Курс будущего» по годам представляет собой геометрическую прогрессию $\{b_n\}$. При этом имеет место соотношение:

$$\frac{2b_{n-1}^2 + 6b_{n-1}b_{n+1} - b_{n+1}^2}{b_{n-1}^2 + b_{n-1}b_{n+1}} = 2.$$

Найти на сколько процентов в год увеличивается капитал Александра Максимовича.

Решение

Пусть знаменатель прогрессии равен q . Тогда условие можно переписать в виде:

$$\frac{2b_{n-1}^2 + 6b_{n-1}^2q^2 - b_{n-1}^2q^4}{b_{n-1}^2 + b_{n-1}^2q^2} = 2 \Rightarrow \frac{2 + 6q^2 - q^4}{1 + q^2} = 2.$$

Пусть $q^2 = t \geq 0$. Имеем:

$$\frac{2 + 6t - t^2}{1 + t} = 2 \Rightarrow t^2 - 4t = 0 \Rightarrow t_1 = 0, t_2 = 4.$$

С учетом того, что прогрессия возрастающая, получаем, что $q = 2$. Поэтому капитал увеличивается на 100%.

Ответ: 100.

Критерии оценивания

1. Обосновано получен верный ответ — 30 баллов.
2. Получены неверные ответы из-за вычислительной ошибки, но при этом имеется верная последовательность всех шагов решения — 20 баллов.
3. Задача верно сведена к биквадратному уравнению — 10 баллов.
4. Решение не соответствует ни одному из критериев, перечисленных выше — 0 баллов.

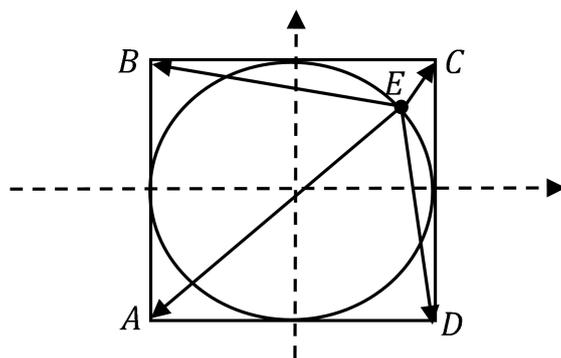
Задача VI.1.2.4. (40 баллов)

Темы: планиметрия, вектора.

Зал обслуживания клиентов банка — квадратный, длина стены равна 10,11 м. Посетители располагаются внутри на удобных диванах, образующих круг. Центральный диван относительно каждой стороны зала касается стены. Посетитель Пафнутий Серафимович Чижииков приходит в зал и занимает место. Он должен решить четыре вопроса: по одному с каждым из менеджеров, сидящих в углах зала. Время ожидания начала обслуживания прямо пропорционально длине вектора, который есть сумма путей, которые Пафнутию Серафимовичу надо пройти от своего места до всех менеджеров (1 м = 1 мин, в одну сторону). Сколько всего минут пришлось ждать Пафнутию Серафимовичу, если предположить, что каждый раз он идет от своего места до соответствующего менеджера по прямой в определенном направлении?

Решение

Зал представляет собой квадрат со стороной 10,11 (например, $ABCD$); диваны — вписанный в него круг, касающийся его сторон. Занятое Чижииковым место обозначим точкой E . Таким образом, задача сводится к нахождению длины вектора: $\vec{EA} + \vec{EB} + \vec{EC} + \vec{ED}$.



Введем декартову систему координат с центром в центре квадрата (и круга). Тогда введенные точки будут иметь следующие координаты:

$$E(x; y) \quad A(-5, 055; -5, 055) \quad B(-5, 055; 10, 055) \quad C(5, 055; 5, 055) \quad D(5, 055; -5, 055).$$

Поэтому:

$$\vec{EA} = (-5, 055 - x; -5, 055 - y) \quad \vec{EB} = (-5, 055 - x; 5, 055 - y)$$

$$\overrightarrow{EC} = (5,055 - x; 5,055 - y) \quad \overrightarrow{ED} = (10,055 - x; -5,055 - y)$$

А значит (в силу того, что точка E лежит на окружности):

$$\overrightarrow{EA} + \overrightarrow{EB} + \overrightarrow{EC} + \overrightarrow{ED} = (-4x; -4y) \Rightarrow |\overrightarrow{EA} + \overrightarrow{EB} + \overrightarrow{EC} + \overrightarrow{ED}| = 4\sqrt{x^2 + y^2} = 20,22.$$

Ответ: 20,22 мин.

Критерии оценивания

1. Обосновано получен верный ответ — 40 баллов.
2. Получены неверные ответы из-за вычислительной ошибки, но при этом имеется верная последовательность всех шагов решения — 30 баллов.
3. Задача верно сведена к нахождению длины суммы векторов — 20 баллов.
4. Есть существенные ошибки, но прослеживается логика решения — 10 баллов.
5. Решение не соответствует ни одному из критериев, перечисленных выше — 0 баллов.

Математика. 10–11 классы

Задача VI.1.3.1. (20 баллов)

Темы: исследование функции с помощью производной.

Функция издержек малого предприятия «Мадагаскарский Таракан» Жуковского Семён Семёновича имеет вид:

$$f(x) = 3x - \frac{4}{\pi} \cdot \sin(\pi x) + \frac{1}{2\pi} \cdot \sin(2\pi x) + 1,$$

где x — количество кг корма для экзотических животных, закупаемого Семён Семёновичем. В среднем обитатели «Мадагаскарского Таракана» потребляют от 674 до 675 кг корма еженедельно. Минимизируйте издержки предпринимателя Жуковского за год, учитывая, что в 2023 году 52 недели.

Решение

Семён Семенычу требуется найти минимум издержек при заданном потреблении. Рассмотрим заданную функцию издержек и исследуем ее на монотонность с помощью производной:

$$\begin{aligned} f'(x) &= \left(3x - \frac{4}{\pi} \cdot \sin(\pi x) + \frac{1}{2\pi} \cdot \sin(2\pi x) + 1\right)' = 3 - 4 \cos(\pi x) + \cos(2\pi x) = \\ &= 3 - 4 \cos(\pi x) + 2 \cos^2(\pi x) - 1 = 2 - 4 \cos(\pi x) + 2 \cos^2(\pi x) = 2(1 - \cos(\pi x))^2 \geq 0. \end{aligned}$$

Заметим, что производная исходной функции всюду неотрицательна, значит сама функция не убывает. В частности, и на заданном отрезке $x \in [674; 675]$. И поэтому принимает свое наименьшее значение в начале отрезка, то есть в точке $x = 674$. Вычислим значение исходной функции в этой точке:

$$f(674) = 3 \cdot 674 - \frac{4}{\pi} \cdot \sin(\pi \cdot 674) + \frac{1}{2\pi} \cdot \sin(2\pi \cdot 674) + 1 = 2023.$$

Полученное значение соответствует минимальной количественной оценке издержек на еженедельное потребление корма. Теперь найдем минимальную оценку издержек на предстоящий год: $2023 \cdot 52 = 105196$.

Ответ: 105196.

Критерии оценивания

1. Обосновано получен верный ответ — 20 баллов.
2. Верно определено поведение функции на отрезке, но получены неверные ответы из-за вычислительной ошибки — 10 баллов.
3. Решение не соответствует ни одному из критериев, перечисленных выше — 0 баллов.

Задача VI.1.3.2. (20 баллов)

Темы: исследование функции с помощью производной, задачи с параметром.

Функция спроса на сувенирное изделие «Символ года — Черный Кролик» (в зависимости от цены) имеет вид:

$$f(x) = |\ln(x - 2)|.$$

Функция предложения товара изготовителем включает в себя параметр стоимости сырья (a) и имеет следующий вид:

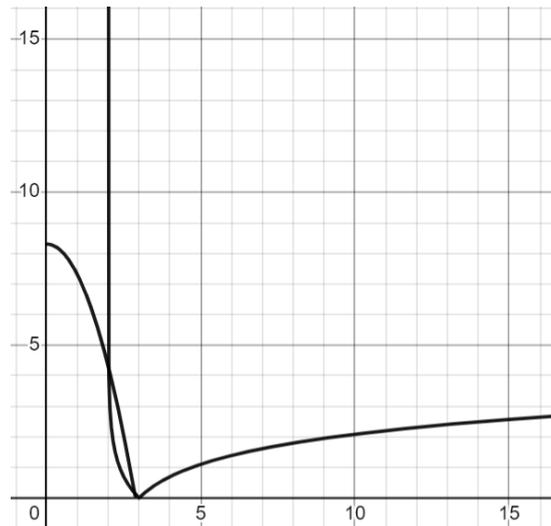
$$g(x) = a - x^2$$

. Найти такой вариант стоимости сырья, при котором обеспечивается равновесный рынок единственно возможным способом.

Примечание: считать, что рынок находится в равновесии, когда спрос равен предложению.

Решение

Построим график функции спроса $f(x) = |\ln(x - 2)|$. Заметим, что область определения этой функции: $x > 2$. Откуда следует, что минимальная стоимость изделия больше 2. Графиком функции предложения является часть параболы $g(x) = a - x^2$, расположенная в первой координатной четверти (исходя из экономического смысла задачи). Далее, из условия понятно, что речь идет о точке касания двух графиков. Построив их, можно увидеть, что касание параболы, возможно только с частью графика $f(x) = |\ln(x - 2)|$ при $x < 3$.



Определим теперь при каком значении параметра a будет иметь место касание графиков. Обозначим точку касания x_0 и применим свойство касательной. Заметим, что при $x < 3$ функция спроса имеет вид: $f(x) = -\ln(x - 2)$. Поэтому:

$$\begin{cases} f(x_0) = g(x_0) \\ f'(x_0) = g'(x_0) \end{cases} \quad \text{или} \quad \begin{cases} |\ln(x_0 - 2)| = a - x_0^2 \\ \frac{-1}{x_0 - 2} = -2x_0 \end{cases} \quad (\text{VI.1.1})$$

Из второго уравнения системы (VI.1.1) получаем квадратное уравнение: $2x_0^2 - 4x_0 - 1 = 0$. Найдем его корни. Первый корень $x_0 = 1 - \frac{\sqrt{6}}{2}$ не подходит, поскольку не удовлетворяет условию $x > 2$. Второй корень $x_0 = 1 + \frac{\sqrt{6}}{2}$. То есть точкой касания будет точка $x_0 = 1 + \frac{\sqrt{6}}{2}$.

Теперь найдем значение параметра стоимости сырья из первого уравнения системы (VI.1.1):

$$a = \left| \ln \left(\frac{\sqrt{6}}{2} - 1 \right) \right| + \left(1 + \frac{\sqrt{6}}{2} \right)^2.$$

Ответ: $\left| \ln \left(\frac{\sqrt{6}}{2} - 1 \right) \right| + \left(1 + \frac{\sqrt{6}}{2} \right)^2.$

Критерии оценивания

1. Обосновано получен верный ответ — 20 баллов.
2. Задача верно сведена к исследованию взаимного расположения графиков соответствующих функций — 10 баллов.

Задача VI.1.3.3. (30 баллов)

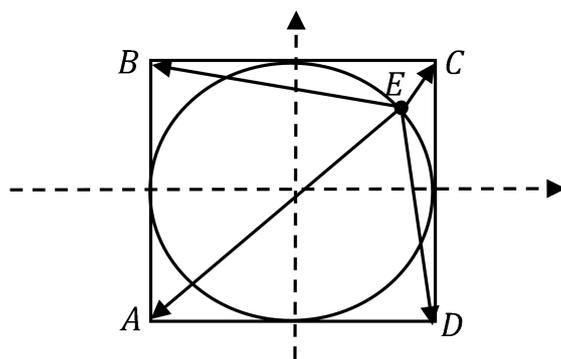
Темы: планиметрия, вектора.

Зал обслуживания клиентов банка — квадратный, длина стены равна 10,11 м. Посетители располагаются внутри на удобных диванах, образующих круг. Центральный диван относительно каждой стороны зала касается стены. Посетитель Пафнутий Семенович Чижигов приходит в зал и занимает место. Он должен решить четыре

вопроса: по одному с каждым из менеджеров, сидящих в углах зала. Время ожидания начала обслуживания прямо пропорционально длине вектора, который есть сумма путей, которые Пафнутию Серафимовичу надо пройти от своего места до всех менеджеров (1 м = 1 мин, в одну сторону). Сколько всего минут пришлось ждать Пафнутию Серафимовичу, если предположить, что каждый раз он идет от своего места до соответствующего менеджера по прямой в определенном направлении?

Решение

Зал представляет собой квадрат со стороной 10,11 (например, $ABCD$); диваны — вписанный в него круг, касающийся его сторон. Занятое Чижиковым место обозначим точкой E . Таким образом, задача сводится к нахождению длины вектора: $\vec{EA} + \vec{EB} + \vec{EC} + \vec{ED}$.



Введем декартову систему координат с центром в центре квадрата (и круга). Тогда введенные точки будут иметь следующие координаты:

$$E(x; y) \quad A(-5, 055; -5, 055) \quad B(-5, 055; 10, 055) \quad C(5, 055; 5, 055) \quad D(5, 055; -5, 055).$$

Поэтому:

$$\vec{EA} = (-5, 055 - x; -5, 055 - y) \quad \vec{EB} = (-5, 055 - x; 5, 055 - y)$$

$$\vec{EC} = (5, 055 - x; 5, 055 - y) \quad \vec{ED} = (10, 055 - x; -5, 055 - y)$$

А значит (в силу того, что точка E лежит на окружности):

$$\vec{EA} + \vec{EB} + \vec{EC} + \vec{ED} = (-4x; -4y) \Rightarrow |\vec{EA} + \vec{EB} + \vec{EC} + \vec{ED}| = 4\sqrt{x^2 + y^2} = 20, 22.$$

Ответ: 20,22 мин.

Критерии оценивания

1. Обосновано получен верный ответ — 40 баллов.
2. Получены неверные ответы из-за вычислительной ошибки, но при этом имеется верная последовательность всех шагов решения — 30 баллов.
3. Задача верно сведена к нахождению длины суммы векторов — 20 баллов.
4. Есть существенные ошибки, но прослеживается логика решения — 10 баллов.
5. Решение не соответствует ни одному из критериев, перечисленных выше — 0 баллов.

Задача VI.1.3.4. (30 баллов)

Темы: бесконечно убывающая геометрическая прогрессия, производные.

Марфа Петровна Кошечкина решила накопить себе некоторую сумму к пенсии. Финансовая организация «Пирамидка» предложила Марфе Петровне следующую схему работы со вкладом в P рублей:

год	сумма вклада на конец года
1	$P^{\frac{1}{P}}$
2	$P^{\frac{1}{P}} \cdot (P^2)^{\frac{1}{P^2}}$
3	$P^{\frac{1}{P}} \cdot (P^2)^{\frac{1}{P^2}} \cdot (P^3)^{\frac{1}{P^3}}$
...	...

Как можно оценить перспективы наращенной суммы Марфы Петровны, если первоначально она планирует отложить 20220 рублей? Следует ли принять предложение «Пирамидки»?

Решение

Очевидно, потенциал Марфы Петровны равен бесконечному произведению:

$$P^{\frac{1}{P}} \cdot (P^2)^{\frac{1}{P^2}} \cdot (P^3)^{\frac{1}{P^3}} \dots$$

Просуммируем степени:

$$\frac{1}{P} + \frac{2}{P^2} + \frac{3}{P^3} + \dots = \sum_{k=1}^{\infty} \frac{k}{P^k}.$$

Пусть $1/P = q$. Тогда последняя сумма примет вид:

$$\sum_{k=1}^{\infty} kq^k.$$

Сумма бесконечно убывающей геометрической прогрессии со знаменателем $q = 1/P$ и первым членом, равным 1, будет равна $1/(1 - q)$. То есть:

$$\sum_{k=0}^{\infty} q^k = \frac{1}{1 - q}.$$

А поскольку $(q^k)' = k \cdot q^{k-1}$, то

$$\left(\sum_{k=0}^{\infty} q^k \right)' = 0 + \sum_{k=1}^{\infty} kq^{k-1} = \frac{1}{(1 - q)^2}.$$

Поэтому:

$$q \cdot \sum_{k=1}^{\infty} kq^{k-1} = \sum_{k=1}^{\infty} kq^k = \frac{q}{(1 - q)^2}.$$

Таким образом, коэффициент наращения средств Марфы Петровны можно оценить следующим образом:

$$20220 \frac{20220}{20219^2} \approx 1.$$

Ответ: 20220 руб, предложение принимать не стоит.

Критерии оценивания

1. Обосновано получен верный ответ — 30 баллов.
2. Задача верно сведена к исследованию суммы ряда — 20 баллов.
3. Есть существенные ошибки, но прослеживается логика решения — 10 баллов.
4. Решение не соответствует ни одному из критериев, перечисленных выше — 0 баллов.
5. Рассуждения о свойствах функции, которые ошибочны, либо не приводят к противоречию — 0 баллов.