



Вариант № 1

Оценка выполнения олимпиадной работы (заполняется проверяющим)											
№ задания	1	2	3	4	5	6	7	8	9	10	Σ
Полученный балл	10	7	7	0	20	20					64
I проверка	Подпись проверяющего			Σ баллов прописью	шестьдесят четыре						
	Подпись проверяющего			Σ баллов прописью	шестьдесят четыре						
II проверка	Подпись проверяющего			Σ баллов прописью	шестьдесят четыре						
	Подпись проверяющего			Σ баллов прописью							

№1

Первая формула считает, сколько цифр выходит за пределы 0-10, т.к. она равна 0, то все цифры, которые есть, летят от 1 до 9

Вторая формула считает, сколько цифр среди ячеек, т.к. она равна 4, то всего 4 цифры

Третья формула считает суммарную длину текста в C2, т.к. текстовая ячейка всего одна, то считает длину C2 или D2 (следовательно в одной из них написан текст длины 2)

Так как в следующей формуле деление на 0, мы понимаем, что E2 и F2 совпадают, из ошибки #ЗНАЧ! мы понимаем, что текст находится в D2, так строка подбора начинается с 3 индекса (т.е. слово фдор), мы понимаем, что $F2 = 3$, а ранее мы поняли, что $E2 = F2 = 3$

По следующей формуле мы понимаем, что это индекс вхождения C2 или D2 в слово ГИПНОТИЗЕР



Вариант № 1

так как мы поняли, что слово только одно, и это

D2 и оно имеет длину 2, то в D2 = ПН

Следующая формула говорит, что суммарная длина строк в C2 и D2 + B2 = 4, так как мы знаем, что длина слова C2 = 0, а D2 = 2, то B2 = 2, а последнее это сумма всех цифр из нее понятно, что C2 = 9.

B2 = 2 C2 = 9 D2 = ПН E2 = 3 ~~F2 = 3~~ F2 = 3

КОД: 2 9 ПН 3 3
+ + + + +

10

№2

Начнем расшифровку, когда переходят буквы при действии шифратора с коэф 1, заметим что П переходит в М, Р переходит в К, и переходит в Ё, следовательно при переходе с коэф 1, буква отодвигается на 3 назад по алфавиту, теперь поймем, как движутся знаки препинания (., ; ?)

Рассмотрим слово ПРИВЕТСТВУЮ, видим, что в перешла в ?, следовательно перед а идет ?



Вариант № 1

Рассмотрим слово $тебя$ $\rightarrow$ перешло в $\bullet$, значит
перед вопросом идет $\bullet$.

Значит перед $\bullet$ идет $\bullet$, следовательно
ровно такой: алфавитный порядок, $\bullet$ - ?
~~Далее~~ $и$

Далее поймём, как движутся при коэф -1

Заметим, что $п$ переходит в $р$, $р$ переходит в $с$,
 $и$ переходит в $й$, значит буква или знак движется
на 1 вперед, исходя из этого запишем полином с
данными нами коэф и найдем его, далее т.к. это
линейный полином, запишем такую систему и
найдем a и b

$$\begin{cases} -a + b = 1 \\ a + b = -3 \end{cases} \quad \begin{matrix} b = -1 \\ a = -2 \end{matrix}$$

Следовательно этот полином $f(x) = -2 \cdot x - 1$
исходя из этого построим таблицу

правило
шифрования +
3б
полином +
2б
2б
незнание
ошибки
в алгебре

7б



Вариант № 1

$x = 1$	$x = -1$
з	б
.	в
?	г
.	д
а	е
б	ё
в	ж
г	з
д	и
е	й
ё	к
ж	л
з	м
и	н
й	о
к	п
л	р
м	с
н	т
о	у
п	ф
р	х
с	ц
т	ч
у	ш
ф	щ
х	ъ
ц	ы
ч	э
ш	ю
щ	я
ъ	.
ы	?
э	:
ю	а
я	



Вариант № 1

№3

Решим второе выражение системы
т.к. там логически знак $\wedge$, то можно
записать это в виде системы

$$\begin{cases} 43 \geq x(x+2) \\ 43 \leq (x-1)^2 \end{cases}$$

Решим I:

$$43 \geq x^2 + 2x \leq 43$$

$$x^2 + 2x - 43 \leq 0$$

$$D = 173$$

$$x_{1,2} = \frac{-1 \pm \sqrt{173}}{2}$$

$$x \in \left[\frac{-1 - \sqrt{173}}{2}, \frac{-1 + \sqrt{173}}{2} \right]$$

$$13 < \sqrt{173} < 14 \Rightarrow \frac{-1 - \sqrt{173}}{2} > -7 \wedge \frac{-1 + \sqrt{173}}{2} < 6$$

Решим II:

$$43 \leq (x-1)^2$$

$$x^2 - 2x - 42 \geq 0$$

$$x \in [-1 - \sqrt{43}; 1 + \sqrt{43}]$$

$$6 < \sqrt{43} < 7 \Rightarrow 1 - \sqrt{43} < -5 \wedge 1 + \sqrt{43} > 7$$

$$\text{Таким образом } x \in \left[\frac{-1 - \sqrt{173}}{2}, 1 - \sqrt{43} \right]$$

Из написанного выше max целый x из этого
промежутка равен -6 , подставляя в систему
получаем уравнение $x^2 = 36, 36 < 52 \Rightarrow$



Вариант № 1

$\Rightarrow$ из лшм следует что угодно, поэтому
первое выражение истина

Ответ - 6

N5

```
1 def dfs(x, y):
2     global n, m, marker, graf, mask, color
3     marker[x][y] = color
4     mask[-1].append([x, y])
5     for dx, dy in [[-1, 0], [1, 0], [0, 1], [0, -1]]:
6         new_x = x + dx
7         new_y = y + dy
8         if new_x < 0 or new_x >= m or new_y < 0 or new_y >= m:
9             continue
10        if marker[new_x][new_y] != color and graf[new_x][new_y] == '#':
11            dfs(new_x, new_y)
12    n, m = [int(s) for s in input()input().split()]
13    graf = [['' for i in range(m)] for j in range(n)]
14    for i in range(n):
15        s = input()input()
16        for j in range(m):
17            graf[i][j] = s[j]
18    mask = []
19    marker = [[0 for i in range(m)] for j in range(n)]
```



Вариант № 1

```
20 color = 1
21 for x in range(n):
22     for y in range(m):
23         if marker[x][y] == 0 and graf[x][y] == '#':
24             mask += [[0]]
25             dfs(x, y)
26             color += 1
27 min_dist = 10 ** 8
28 for point1 in mask[0]:
29     for point2 in mask[1]:
30         dist = abs(point1[0] - point2[0]) + abs(point1[1] - point2[1])
31         if dist < min_dist:
32             min_dist = dist
33 print(min_dist - 1)
```

20

200 - ?

№6

```
def move(x, y):
    global area, marker, dx_dy, direction, n, m, graf
    if marker[x][y] == 0:
        area += 1
        marker[x][y] += 1
        new_x = x + dx_dy[direction][0]
        new_y = y + dx_dy[direction][1]
        if new_x < 0 or new_y < 0 or new_x >= n or new_y >= m:
            return None
        if graf[new_x][new_y] == '#' and marker[new_x][new_y] < 4:
            direction = (direction + 1) % 4
            move(new_x, new_y)
```

20

200 - ?



Вариант № 1

```
elif marker[new-x][new-y] < 4:  
    move(new-x, new-y)
```

```
n, m = [int(s) for s in input().split()]  
graf = [['' for i in range(m)] for j in range(n)]
```

```
for i in range(n):
```

```
    s = input()
```

```
    for j in range(m):
```

```
        graf[i][j] = s[j]
```

```
dx-dy = {0: [1, 0], 1: [0, -1], 2: [-1, 0], 3: [0, 1]}
```

```
max-area = -10 ** 9
```

```
best-point = [0, 0, 0]
```

```
for x in range(n):
```

```
    for y in range(m):
```

```
        if graf[x][y] == '#':
```

```
            continue
```

```
        for direction in dx-dy:
```

```
            area = 0
```

```
            marker = [['0' for i in range(m)] for j in range(n)]
```

```
            move(x, y)
```

```
            if max-area < area:
```

```
                max-area = area
```

```
                best-point = [x, y, direction]
```

```
print(max-area)
```




Олимпиада школьников «Гранит науки»
БЛАНК ОЛИМПИАДНОЙ РАБОТЫ

Шифр работы МЗ4-161
(не заполнять)

Вариант № 1

№	Название	Число хостов	Адрес	Макс. прер.	десятич. маска	Шлюз	диапазон
	Главный корпус	1000					
	Инженерный корпус	500					
	Учебный центр №2	750					
	Библиотека	30					
	Посты охраны	14					
	IT-отдел	4					

об
таблица
не заполнена

