



Вариант № 2

Оценка выполнения олимпиадной работы
(заполняется проверяющим)

№ задания	1	2	3	4	5	6	7	8	9	10	Σ
Полученный балл	-	10	10	-	20	20					60
I проверка	Подпись проверяющего				Σ баллов прописью	шестьдесят					
	Подпись проверяющего				Σ баллов прописью	шестьдесят					
II проверка	Подпись проверяющего				Σ баллов прописью	шестьдесят					
	Подпись проверяющего				Σ баллов прописью						

Задача № 3.

$$\begin{cases} 12 \geq x^2 \rightarrow 12 \leq (x-1)^2 \\ 26 \geq x(x+3) \leftrightarrow 26 \leq (x^2 + 2x + 1) \end{cases}$$

Знак импликации ($\rightarrow$) в первой строке означает, что

~~для~~ если x является решением левой части, то он также является и решением правой. При этом, если неравенство левое неверно, то правое должно быть верным.

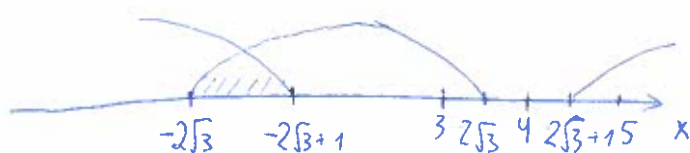
$$12 \geq x^2$$
$$-2\sqrt{3} \leq x \leq 2\sqrt{3}$$

$$12 \leq (x-1)^2$$

$$\begin{cases} x-1 \geq 2\sqrt{3} \\ x-1 \leq -2\sqrt{3} \end{cases}$$

$$\begin{cases} x \geq 2\sqrt{3} + 1 \\ x \leq -2\sqrt{3} + 1 \end{cases}$$

расположим полученные значения на числовой прямой.



108



Вариант № 2

$$\sqrt{3} \approx 1,6$$

$$2\sqrt{3} \approx 3,2$$

так как нам подходит все случаи, кроме " $1 \rightarrow 0$ ",
то возмозможны решениями будут числа $x \geq 4$
~~то~~ (малые мы не берем т.к. нужно найти наибольшее,
а 3 уже не подходит, т.к. экв. решением левой части (1)
и не лвл. решением второй (0)).

Обратимся к второй строке. Нам подходит случаи, когда
либо оба неравенства верны, либо оба неверны.

$$26 \geq x(x+3)$$

$$26 \leq (x^2 + 2x + 1)$$

$$x^2 + 3x - 26 \leq 0$$

$$26 \leq (x+1)^2$$

$$D = 9 + 26 \cdot 4 = 9 + 104 = 113$$

$$x = \frac{-3 \pm \sqrt{113}}{2}$$

$$\begin{cases} x+1 \geq \sqrt{26} \\ x+1 \leq -\sqrt{26} \end{cases}$$

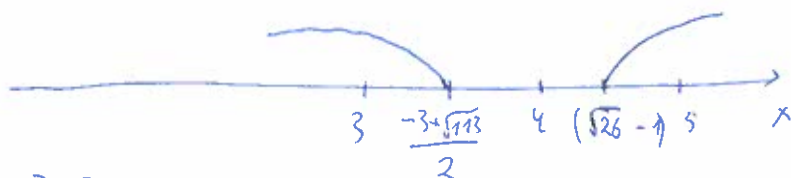
$$10 < \sqrt{113} < 11$$

$$\begin{cases} x \geq \sqrt{26} - 1 \\ x \leq -\sqrt{26} - 1 \end{cases}$$

$$3,5 < \frac{-3 + \sqrt{113}}{2} \leq 4 \quad -7 < \frac{-3 - \sqrt{113}}{2} < -6,5$$

$$\sqrt{26} > 5$$

$$\sqrt{26} - 1 > 4$$



(слева числа ≤ 3 не наносим, т.к. они не нужны)

Из рисунка видно, что числа, большие 4 (большие $\sqrt{26}-1$)
подходят только одному неравенству.
Значит, наше искомое число = 4. (0 $\leftrightarrow$ 0 - верно)



Вариант № 2

выполним проверку для $x = 4$. (0 - ложь, 1 - истина)

$$\begin{cases} 12 \geq 4^2 \rightarrow 12 \leq (4-1)^2 \\ 26 \geq 4(4+3) \leftrightarrow 26 \leq (4^2 + 2 \cdot 4 + 1) \end{cases}$$

$$\begin{cases} 0 \rightarrow 0 & \text{- верно} \\ 0 \leftrightarrow 0 & \text{- верно} \end{cases}$$

Ответ: 4 прав.

Задание 4.

название подсети	число хостов в сети	адрес подсети	маска - префикс	десятичная маска	илиз по умалчанию	диапазон доступных адресов для узлов
филиал	2000		121	255.255.248.0	+	
производ. корпуса	1000		122	255.255.252.0	+	
научно- конструктор- ские отделы	500		123	255.255.254.0	+	
инженерия	30		127	255.255.255.224	-	

05



Вариант № 2

Задание 2.

Видно, что шифр представляет собой шифр Цезаря со сдвигом $kx + b$, где x - введенное число. таблица соответствий (буквы и символы не обязательно по порядку)

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

при $x = 1$ разница равна -3

при $x = -1$ разница равна 5

$$\Rightarrow \begin{cases} k = -4 \\ b = 1 \end{cases}$$

получаем сдвиг $-4x + 1$

Ответ: шифр Цезаря со сдвигом $-4x + 1$

част. решено алг.

200

Задание 5.

20

алгоритм будет состоять из 2 частей. сначала необходимо разделить точки на два острова, а затем для береговых проверить минимальное расстояние до другого острова.

необходимое количество клеток вычисляется по формуле $(\Delta x + \Delta y - 1)$, где Δx и Δy - расстояния между клетками, спроецированными на оси x и y . (горизонтальные и вертикальные расстояния).

$$\Delta x = |x_2 - x_1|; \Delta y = |y_2 - y_1|$$



Вариант № 2

в основной библиотеке

код на языке python

from math import * # необходимо ТОЛЬКО если нет функции abs

s1 = []

s2 = []

f1, f2 = 0, 0

map_ = []

arr = []

x = 30 # ширина карты

y = 15 # длина карты

for i in range(y + 2)

arr.append(input())

q = [0] * (x + 2)

map_.append(q)

for i in range(y):

for j in range(x):

p = arr[i][j]

if p == '~':

pass

elif p == '#':

elif f1 == 0:

f1 = 1

map_[i+1][j+1] = 1

elif f2 == 0:

f2 = 1

map_[i+1][j+1] = 2

elif (map_[i][j+1] == 1) or (map_[i+1][j+2] == 1) or (map_[i+1][j] == 1) or (map_[i+2][j+1] == 1):

map_[i+1][j+1] = 1

elif (map_[i][j+1] == 2) or (map_[i+1][j+2] == 2) or (map_[i+1][j] == 2) or (map_[i+2][j+1] == 2):

map_[i+1][j+1] = 2



Вариант № 2

распределяем точки по массивам s1 и s2

каждая точка задается двумя координатами

```
for i in range(y+1):
```

```
    for j in range(x+1):
```

```
        if map[i][j] == 1:
```

```
            s1.append([i, j])
```

```
        elif map[i][j] == 2:
```

```
            s2.append([i, j])
```

находим минимальное расстояние

$m = x + y + 4$

```
for i in s1:
```

```
    for j in s2:
```

```
        if m > abs(x s1[0] - s2[0]) + abs(s1[1] - s2[1]) - 1:
```

```
            m = abs(s1[0] - s2[0]) + abs(s1[1] - s2[1]) - 1
```

```
print("ответ: ", m)
```

если дополнительно в строки, отсечение
после цифр 1 и 2 условие " and (map[i][j] + map[i-1][j] + map[i][j-1] +
дописать map[i][j+1]) " (и если 1 и в если 2)
программа будет анализировать только приростные области.
В результате время выполнения в худшем случае
(с острова - 3 км) будет приближаться к $O(n^2)$



Вариант № 2

```
def f(arr, x, y, k):  
    m=0  
    if k == 0:  
        while arr[x][y] != 0:  
            m += 1  
            x -= 1  
    elif k == 1:  
        while arr[x][y] != 0:  
            m += 1  
            y += 1  
    elif k == 2:  
        while arr[x][y] != 0:  
            m += 1  
            x += 1  
    else:  
        while arr[x][y] != 0:  
            m += 1  
            y -= 1  
    return m - 1
```

функция подсчета линии
движения *

l_m = 0