



Вариант № 1

Оценка выполнения олимпиадной работы (заполняется проверяющим)											
№ задания	1	2	3	4	5	6	7	8	9	10	Σ
Полученный балл	0	0	10	0	20	0	0	0	0	0	30
I проверка	Подпись проверяющего			Σ баллов прописью							
	Подпись проверяющего			Σ баллов прописью							
II проверка	Подпись проверяющего			Σ баллов прописью		тридцать					
	Подпись проверяющего			Σ баллов прописью		тридцать					

7



Вариант № 1

$$3. \begin{cases} 52 \leq x^2 \rightarrow 52 \geq (x^2 + 1 - 2x) \\ 43 \geq x(x+2) \wedge 43 \leq (x-1)^2 \end{cases}$$

a	b	a+1	a-b
0	0	0	1
0	1	0	1
1	0	0	0
1	1	1	1

60

$$\begin{cases} 43 \geq x^2 + 2x \Rightarrow x^2 + 2x - 43 \leq 0 \Rightarrow x \in [-2\sqrt{11} - 1, 2\sqrt{11} - 1] \\ 43 \leq (x-1)^2 \Rightarrow x^2 - 2x - 42 \geq 0 \Rightarrow x \in (-\infty, 1 - \sqrt{43}] \cup [1 + \sqrt{43}, +\infty) \end{cases}$$

ТЕПЕРЬ НАДО ИХ ОБЪЕДИНИТЬ

$$\sqrt{11} \approx 3,32 \quad \sqrt{43} \approx 6,56$$

$$1 - \sqrt{43} \approx -5,56 \quad 1 + \sqrt{43} \approx 7,56 \quad -2\sqrt{11} - 1 \approx -7,64 \quad 2\sqrt{11} - 1 \approx 5,64$$

Получаем

$$x \in [-7,64; 5,64] \text{ и } x \in (-\infty; -5,56] \cup [7,56; +\infty)$$

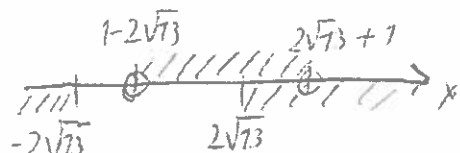
А ТАКЖЕ

$$x \in [-2\sqrt{11} - 1; 1 - \sqrt{43}] \approx [-7,64; -5,56]$$

Мы можем заметить, что $a \rightarrow b = 0$ только при $a=1$ и $b=0$. Поэтому
завяжем найдем множество T корней системы $\begin{cases} 52 \leq x^2 \\ 52 < (x^2 + 1 - 2x) \end{cases}$

А потом найдем множество X корней $52 \leq x^2 \rightarrow 52 \geq (x^2 + 1 - 2x)$,
как $X = (-\infty; +\infty) \setminus T$

$$\begin{cases} x^2 \geq 52 \Rightarrow x \in (-\infty; -2\sqrt{13}] \cup [2\sqrt{13}; +\infty) \\ x^2 - 2x - 51 \geq 0 \Rightarrow \text{н/д} \end{cases}$$



$$\text{Итого: } x \in [2\sqrt{13}; 2\sqrt{13} + 1)$$

но это T

$$\text{Тогда } X = (-\infty; 2\sqrt{13}) \cup [2\sqrt{13} + 1; +\infty)$$



Вариант № 1

Теперь надо объединить $(-\infty; 2\sqrt{3}) \cup [2\sqrt{3}+1; +\infty)$ и $[-2\sqrt{11}-1; 1-\sqrt{43}]$

$$\sqrt{13} \approx 3,61$$

Поэтому

$$x \in [-7,64; -5,56]$$

$$x = -6$$

Ответ: $x = -6$

5. Алгоритм ТРЮВ:

- 1) Записываем все участки суши
- 2) Находим 2 острова
- 3) Находим у 2 островов ближайшие точки
- 4) Вычисляем расстояние между ними

Возможные случаи:

- У нас 0 островов ($dist=0$)
- У нас 1 остров ($dist=1$)
- У нас 2 острова ($dist=x$)

Код программы (на ANSI C) (89/80):

```
#include <stdio.h>
#include <stdlib.h>
#define WATER 0
#define ISLAND1 1
#define ISLAND2 2
#define NOTWATER ISLAND2 // это сделано (целую не заполните лишней пробелом
// это замена count2 - countland - просто так.
#define H 25 // ширина
#define W 30 // высота
char map[H][W];
int zisland1x, zisland1y, zisland2x, zisland2y;
int neighbours1(int x, int y) {
    if (y > 0 && map[y-1][x] == ISLAND1)
        return y > 0 && map[y-1][x] || y < H-1 && map[y+1][x] == ISLAND1 ||
        x > 0 && map[y][x-1] == ISLAND1 || x < W-1 && map[y][x+1] == ISLAND1;
}
```



```

}
int main() {
    int i, j, countLand, countI, mindist, dist;
    for (i = countLand = 0; i < H; i++) {
        for (j = 0; j < W; j++) {
            if (getchar() == '\n') {
                map[i][j] = WATER;
            } else {
                map[i][j] = NOT WATER;
                countLand++;
            }
        }
        getchar(); // не АБВ \n\n
    }
    if (!countLand) { // у нас 0 островов
        putchar('0');
        putchar('\n');
        return 0;
    }
    for (i = 0; i < H; i++) {
        for (j = 0; j < W; j++) {
            if (map[i][j] == NOT WATER) {
                map[i][j] = ISLAND;
                break;
            }
        }
    }
    for (i = countI = 0; i < H; i++) {
        for (j = 0; j < W; j++) {
            if (map[i][j] == NOT WATER && !neighbours(i, j)) {

```



Вариант № 1

20

```
map[i][j] = ISLAND1;
count1++;
}
}
}
if (!count2) count2 = countland - count1;
if (!count2) { // 7 КАС 1 ОСТРОВЕ/
    putchar('0');
    putchar('\n');
    return 0;
}
mindist = w + h - 1;
islandx = malloc(sizeof(int) * count2); // ВОЗМОЖЕ ЗДЕСЬ НУЖНО ДЕЛАТЬ ПРОВЕРКУ НА
islandy = malloc(sizeof(int) * count2); // NULL, но в любом случае при ошибке ВОЗЛЕ
islandx = malloc(sizeof(int) * count1); // ПЕРИМЕТРА НАША ПРОГРАММА ДАЛЬШЕ НЕ П/
islandy = malloc(sizeof(int) * count1); // СМОЖЕТ РАБОТАТЬ/
count1 = count2 = 0;
for (i = 0; i < h; i++) {
    for (j = 0; j < w; j++) {
        if (map[i][j] == ISLAND1) {
            islandx[count1] = j;
            islandy[count1++] = i;
        } else if (map[i][j] == ISLAND2) {
            islandx[count2] = j;
            islandy[count2++] = i;
        }
    }
}
for (i = 0; i < count1; i++) {
    for (j = 0; j < count2; j++) {
```



Вариант № 1

```
dist = abs(island1x[i] - island2x[j]) + abs(island1y[i] - island2y[j]) - 1;
if (dist < mindist) {
    mindist = dist;
}
```

```
}
}
}
printf("%d\n", mindist);
free(island1x);
free(island2x);
free(island1y);
free(island2y);
return 0;
}
```

$$1. B_4 = (ЧЕТЕ(ММ(B_2=F_2; " >= 10")) + (ЧЕТЕ(ММ(B_2=F_2; " <= 0"))$$

$$B_5 = ЧЕТ(B_2=F_2)$$

$$B_6 = \text{Л}(ТР(ЕСМ(ЕТЕК(С_2); (2; ТЕК(С_2; "#")))) + \text{Л}(ТР(ЕСМ(ЕТЕК(С_2); (2; ТЕК(С_2; "#"))))$$

$$B_7 = (УММ(B_2=D_2) / (Е_2 - F_2))$$

$$B_8 = \$D_2 + E_2 + F_2$$

$$B_9 = П(ТР("ПОДБОР"; $F_2; 6))$$

$$B_{10} = \text{ПОИСК}(ЦЕНИТЬ(ЕСМ(ЕТЕК(С_2); (2; "")); ЕСМ(ЕТЕК(С_2); (2; "")); "ПРИНТИЗЕР"))$$

$$B_{11} = \text{Л}(ТР(ЦЕНИТЬ(ЕСМ(ЕТЕК(С_2); (2; "")); ЕСМ(ЕТЕК(С_2); (2; "")); D_2; D_2; "")) + B_2)$$

$$B_{12} = (УММ(B_2=F_2))$$

В(ЕЧИСЛА < 10 и > 0 В(ЕТО ЧИСЛО
Е - Е = 0 И - ИРОД