

## Задача 1

9) #include <iostream>  
#include <vector>  
using namespace std;

```
int main() {
    vector<int> data(32, 255);
    vector<int> lost(32, 1000);
    string cmd;
    while (true) {
        cin >> cmd;
        if (cmd == "add") {
            int i = 0;
            if (data[25] == 255) {
                while (i < 30 && lost[i] < 1000) i += 3;
            } else {
                while (i < 30 && lost[i] < 1000) i += 3;
            }
            int h, m, t;
            cout << "Введите номер цикла в век" << endl;
            cin >> h;
            cout << "Введите номер цикла в массиве" << endl;
            cin >> m;
            cout << "Введите номер" << h << ":" << m << ":";
            cout << "Введите значение элемента в заданном месте" << endl;
            cin >> t;
            data[i] = h;
            data[i+1] = m;
            data[i+2] = t;
            lost[i]--;
        }
        if (cmd == "all") {
            int i = 0;
            while (i < 30 && data[i] != 255) { cout << data[i] << " "; cout << data[i+1] << " "; cout << data[i+2] << " "; cout << "T=" << data[i+2] << endl; }
        }
        if (cmd == "cycles") {
            int i;
            cout << "Введите номер цикла" << endl;
            cin >> i;
            cout << "Средняя" << i << " форма перемещения" << 1000 - lost[i-i%3] << " раз" << endl;
        }
    }
}
```

// C++

Задача 1:  
1. Создать вектор "data" и  
2. Добавить элемент в конец  
3. Добавить элемент в конец  
4. Добавить элемент в конец  
5. Добавить элемент в конец  
6. Добавить элемент в конец  
7. Добавить элемент в конец  
8. Добавить элемент в конец  
9. Добавить элемент в конец  
10. Добавить элемент в конец  
11. Добавить элемент в конец  
12. Добавить элемент в конец  
13. Добавить элемент в конец  
14. Добавить элемент в конец  
15. Добавить элемент в конец  
16. Добавить элемент в конец  
17. Добавить элемент в конец  
18. Добавить элемент в конец  
19. Добавить элемент в конец  
20. Добавить элемент в конец  
21. Добавить элемент в конец  
22. Добавить элемент в конец  
23. Добавить элемент в конец  
24. Добавить элемент в конец  
25. Добавить элемент в конец  
26. Добавить элемент в конец  
27. Добавить элемент в конец  
28. Добавить элемент в конец  
29. Добавить элемент в конец  
30. Добавить элемент в конец  
31. Добавить элемент в конец  
32. Добавить элемент в конец  
33. Добавить элемент в конец  
34. Добавить элемент в конец  
35. Добавить элемент в конец  
36. Добавить элемент в конец  
37. Добавить элемент в конец  
38. Добавить элемент в конец  
39. Добавить элемент в конец  
40. Добавить элемент в конец  
41. Добавить элемент в конец  
42. Добавить элемент в конец  
43. Добавить элемент в конец  
44. Добавить элемент в конец  
45. Добавить элемент в конец  
46. Добавить элемент в конец  
47. Добавить элемент в конец  
48. Добавить элемент в конец  
49. Добавить элемент в конец  
50. Добавить элемент в конец  
51. Добавить элемент в конец  
52. Добавить элемент в конец  
53. Добавить элемент в конец  
54. Добавить элемент в конец  
55. Добавить элемент в конец  
56. Добавить элемент в конец  
57. Добавить элемент в конец  
58. Добавить элемент в конец  
59. Добавить элемент в конец  
60. Добавить элемент в конец  
61. Добавить элемент в конец  
62. Добавить элемент в конец  
63. Добавить элемент в конец  
64. Добавить элемент в конец  
65. Добавить элемент в конец  
66. Добавить элемент в конец  
67. Добавить элемент в конец  
68. Добавить элемент в конец  
69. Добавить элемент в конец  
70. Добавить элемент в конец  
71. Добавить элемент в конец  
72. Добавить элемент в конец  
73. Добавить элемент в конец  
74. Добавить элемент в конец  
75. Добавить элемент в конец  
76. Добавить элемент в конец  
77. Добавить элемент в конец  
78. Добавить элемент в конец  
79. Добавить элемент в конец  
80. Добавить элемент в конец  
81. Добавить элемент в конец  
82. Добавить элемент в конец  
83. Добавить элемент в конец  
84. Добавить элемент в конец  
85. Добавить элемент в конец  
86. Добавить элемент в конец  
87. Добавить элемент в конец  
88. Добавить элемент в конец  
89. Добавить элемент в конец  
90. Добавить элемент в конец  
91. Добавить элемент в конец  
92. Добавить элемент в конец  
93. Добавить элемент в конец  
94. Добавить элемент в конец  
95. Добавить элемент в конец  
96. Добавить элемент в конец  
97. Добавить элемент в конец  
98. Добавить элемент в конец  
99. Добавить элемент в конец  
100. Добавить элемент в конец

{Продолжение на с. 2}

W 1

```

if (cmd == "test") {
    // int m = 1000;
    // for (int i = 0; i < 30; i += 3) m = min(m, data[i]);
    cout << m << endl;
    int sum = 0;
    // for (int i = 0; i < 30; i += 3) sum += data[i];
    for (int i = 0; i < 30; i += 3) sum += max(0, data[i] - 1);
    cout << "Внепронзабученный набор (задание 1) "
        << sum << " значение" << endl;
}

```

```

return 0;
}

```

Кер-то все равно равно нулю  
значение (каждое значение  
запускает -1), но надо  
считать модор не число -1.

10 #include <bits/stdc++.h>  
using namespace std;

// C++  
Анонимное поле उपयोग

25 int main() {  
 vector<vector<int>> a(8, vector<int>(8, 0));  
 string cmd;  
 // while  
 int distance;  
 while (true) {  
 if (cmd == "add") {  
 // vector add  
 vector<pair<int, int>> add(0);  
 for (int i = 0; i < 7; ++i) {  
 for (int j = 0; j < 7; ++j) {  
 if (a[i][j] == 0 && a[i+1][j+1] == 0 && a[i+1][j] == 0  
 && a[i][j+1] == 0) {  
 add.push\_back(make\_pair(i, j));  
 }  
 }  
 }  
 }  
 }  
}

```

if (add.size() == 0) { cout << "Керто додано" << endl; }
else {

```

```

    random_shuffle(add.begin(), add.end());
    a[add[0].first][add[0].second] = 8;
    a[add[0].first+1][add[0].second] = 8;
    a[add[0].first][add[0].second+1] = 8;

```

comp 3  
✓

N 2

```

    a[add[0].first + i][add[0].second + 1] = 8;
    cout << "Punya bobobene" << endl;
}
if (cmd == "distance") {
    int i;
    cout << "Abendne Distanz" << endl;
    cin >> i;
    distance = ceil(((float)i * 8) / 100);
    cout << "Distanz: " << distance << " px" << endl;
}
if (cmd == "left") {
    for (int i = 0; i < 8; i++) {
        for (int j = 1; j < 8; j++) {
            if (a[i][j-1] >= distance) { a[i][j-1] += distance;
                a[i][j] -= distance; }
            a[i][j] = max(0, a[i][j] - distance);
        }
    }
    cout << "Punya obengene beke" << endl;
}
if (cmd == "right") {
    for (int i = 0; i < 8; i++) {
        for (int j = 0; j < 7; j++) {
            if (a[i][j+1] >= distance) { a[i][j+1] += distance;
                a[i][j] -= distance; }
            a[i][j] = max(a[i][j] - distance, 0);
        }
    }
    cout << "Punya obengene bupabo" << endl;
}
if (cmd == "up") {
    for (int i = 1; i < 8; i++) {
        for (int j = 0; j < 8; j++) {
            if (a[i-1][j] >= distance) { a[i-1][j] += distance;
                a[i][j] -= distance; }
            a[i][j] = max(0, a[i][j] - distance);
        }
    }
    cout << "Punya obengene bupx" << endl;
}

```

Opodomenne na step 4.

w 3

```

if (cmd == "down") {
    for (int i = 0; i < 8; ++i) {
        for (int j = 0; j < 8; ++j) {
            if (a[i+1][j] == distance) { a[i+1][j] += distance;
                a[i][j] = max(0, a[i][j] - distance);
            }
        }
    }
    cout << "Прогноз движения брызг" << endl;
}

if (cmd == "show") {
    for (int i = 0; i < 8; ++i) {
        for (int j = 0; j < 8; ++j) {
            cout << a[i][j] << " ";
        }
        cout << endl;
    }
}

```

Алгоритм и задание 10.

создадим vector с числами. 8x8  
add() - введем квадрат 2x2 из дощечки и сместим их в "add". Перемешиваем его; берем первые элементы (или Random); → куда и будем возвращать квадрат.

если size(add) → дощечка есть и не

distance - перемешиваем и пропорции, округляем вверх по формуле  $dist = \lceil \frac{8 \cdot x}{100} \rceil$   
 x - процент

left, right и up если left начинаем двигаться с самых лево, если right - с правых и т.д.

и будем перемешивать поочередно вправо. Подвинувшись можно если dist + 100, в среднем куда хотим подвинуться  $\leq 8$   
 Проходим не по всем средним квадрат не уперевшись в стены.

④ Воспользуемся алгоритмом Дейкстры.

⑤ (так и. населенные пункты с дорогами с расстоянием между ними).

Рядом с каждым пунктом.

Будем считать  $dist_i = 0$ .

Составим массив  $dist$  — расстояние от  $A$  до  $i$  и будем пересчитывать значения.

$$dist[j] = \min(dist[i], dist[i] + a[i][j])$$

нон.  
гнор

→  $dist$

|   |                |                |                |                |                 |                |                |                 |                 |                 |                 |
|---|----------------|----------------|----------------|----------------|-----------------|----------------|----------------|-----------------|-----------------|-----------------|-----------------|
| 0 | 4 <sub>A</sub> | 5 <sub>A</sub> | 3 <sub>A</sub> |                |                 |                |                |                 |                 |                 |                 |
| 0 | 4 <sub>A</sub> | 4 <sub>B</sub> | 3 <sub>B</sub> |                |                 | 5 <sub>B</sub> |                |                 |                 |                 |                 |
| 0 | 4 <sub>A</sub> | 4 <sub>B</sub> | 3 <sub>B</sub> | 7 <sub>B</sub> |                 | 5 <sub>B</sub> |                |                 |                 |                 |                 |
| 0 | 4 <sub>A</sub> | 4 <sub>B</sub> | 3 <sub>B</sub> | 6 <sub>C</sub> | 12 <sub>C</sub> | 5 <sub>C</sub> |                |                 |                 |                 |                 |
| 0 | 4 <sub>A</sub> | 4 <sub>B</sub> | 3 <sub>B</sub> | 6 <sub>C</sub> | 10 <sub>C</sub> | 5 <sub>C</sub> |                | 12 <sub>C</sub> | 13 <sub>C</sub> |                 |                 |
|   |                |                |                |                | 8 <sub>E</sub>  | 5 <sub>D</sub> | 8 <sub>E</sub> | 12 <sub>C</sub> | 13 <sub>C</sub> |                 |                 |
|   |                |                |                |                |                 |                | 8 <sub>E</sub> | 9 <sub>F</sub>  | 13 <sub>C</sub> |                 |                 |
|   |                |                |                |                |                 |                |                | 9 <sub>F</sub>  | 11 <sub>H</sub> | 12 <sub>H</sub> |                 |
|   |                |                |                |                |                 |                |                |                 | 11 <sub>H</sub> | 11 <sub>C</sub> |                 |
|   |                |                |                |                |                 |                |                |                 |                 |                 | 16 <sub>D</sub> |

A B C D E F G H I J K  
✓ ✓ ✓ ✓ ✓ ✓ ✓ ✓ ✓ ✓

кратчайший путь = 11

Восстановим путь:

K I F E C D A.

→ A D C E F I K

Ответ: 11 : A D C E F I K

7) Переписать программу в более удобном формате

$$n = 20$$

$$a = [1; 2; 3 \dots 20]$$

$$i = 20; k = 0; m = 10;$$

while ( $a[i] > k$ ):

while ( $a[i-1] \geq m$ ):

swap( $a[i], a[i-1]$ )

$k++$

$i--$

}

$$i = n$$

$$m = a[i] / 2$$

Алг. для нахождения того, что используется  
нечисленное деление!

В первую очередь. Внутр. цикл. продолжается  
за 10 итер.  
20. Внутр. и получаем: 1; 2; ... 9; 20; 10; 11; ... 19.

$$m = \frac{19}{2} = 9.$$

Далее продолжается 19  $\rightarrow$  1... 8; 19; 9; 20... 18.  
За 11 итер.

тогда каждый 2 итерации во внешн. цикле

каждый итерации во внутр. цикле  $\uparrow$  на 2.

↓ последний элемент. 19, 18, 17, 16, 15, 14, 13, 12, 11, 10.  
на сколько  $\uparrow$  к. 10, 11, 11, 12, 12, 13, 13, 14, 14, 15.

$$10 + 11 + 11 + 12 + 12 + 13 + 13 + 14 + 14 + 15 =$$

$$= 125$$

здесь закон.  
программ.  
к.к.  $a[20] = 10$

Ответ:  $K = 125$ ,  $k = 125$ .

106