

Вариант 2

31

номер листа

Задача 17

0

Очевидно, что этот алгоритм будет брать посл-е число и перетаскивать его к 29. Алгоритм зав-се когда посл-е число станет = 30. Всп-и t тр-се кол-во переез.

В конце массив будет вида

$[1, 2, 3, \dots, 29, 31, 32, \dots, 59, 60, 30]$

$\Rightarrow$ произойдет перестановка 30 чисел, каждого на 30 мес.

$\Rightarrow t = 30 \cdot 30 = 900$

Ответ: 900

Задача 15

7

В основе решения этой задачи лежит алг-м разделения и властности или бин. поиска

Шаг 1: Имеем 6 канистр из первых в канистр и вторые в канистр из группы. Потратим 1 полоску для того, чтобы узнать в какой группе зар-е канистр.

Шаг 2: Алг-м разделим группу из 6 канистр на 2 группы по 3 канистры. Потратим 1 полоску для того, чтобы понять в какой группе зар-е канистр.

Шаг 3: Алг-м разделим ост-е канистры на 2 группы: по 1 и по 2 канистры. Потратим 1 полоску для определения в какой группе зар-е. Если зар-е группа с 1ой канистрой, то мы нашли, если нет, то Шаг 4

номер листа (Вариант 3)

2

Шаг 4: Разделим ост-ше группу из 2х канистр на 2 группы по 1 канистре

Потратим 1 полоску для вывешивания в какой именно кан-ре зар. вода

Вывод: так как бинарный поиск имеет лог-ую сложность в худшем случае, мно пом-сь $\log_2 12 = 4^*$ действие

$\log_2 12 \neq 4$, но в progr. $\log$ округляет вверх

Задача №1

Всего 2560 КБ

⇒ Текстовые страницы ≈ 42 КБ

Фотографии ≈ 2518 КБ

⇒ Текстовые страницы: 47 стр

Кол-во картинок: 64

Ответ: III стр

Всё одной карт. =

~~3.230.470~~ = ~~6186000~~ 3245000

$\approx 39,6$ КБ

Всё текста

$460 \cdot 16 = 7360$ Бит

$\approx 0,9$ КБ

Задача №3, (3)

Условие некоректно

В корпсе абзаце написано, что: "Известно, что ~~муж~~ мунегини работают только в основном департаменте"

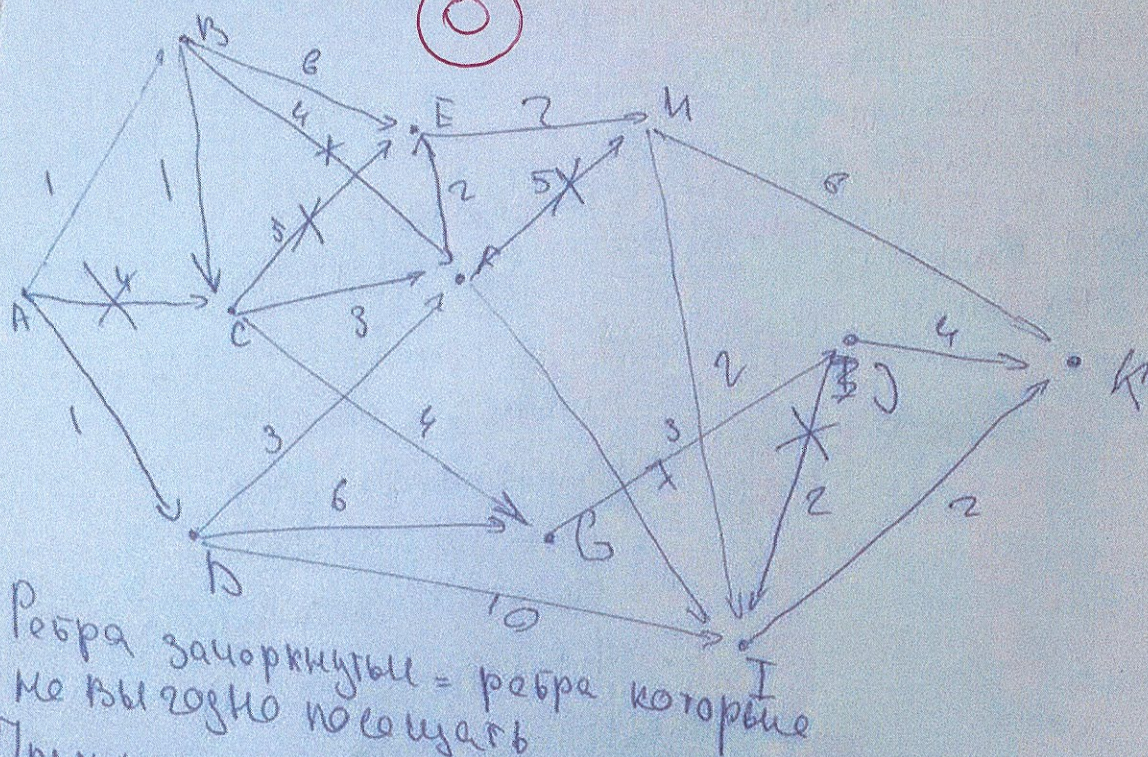
А по пункту 2, написано что кол-во мунегин во вспом. департаменте $\neq 0$. Противор.

Если я всё правильно понял, то прошу аннулировать это задание

Номер листа
3

Задача №4

(Вариант №3)



Ребра зачёркнуты = ребра которые
не выгодно посещать

Применим алгоритм обратной обхода в глубину

Получим пути

K J G C B A - 13

K K E C B A - 15

K M E B A - 15

K I D A - 13

K I F B A - 13

K I F B A - 14

Ответ: K J G C B A - 13

12!

Вариант №3

4

Задача №10

Рог на GNU C++ 17

```

#include <bits/stdc++.h>
using namespace std;
vector<vector<int>> t(8, vector<int>(8, 0));
int n = 8; int m = 8; vector<pair<int, int>> f;
void add() {
    for (int i = 0; i < 10; i++) {
        pair<int, int> e = {rand() % (n-1)+1, rand() % (m-2)+1};
        if (t[e.first][e.second] == 0 && t[e.first+1][e.second] == 0 && t[e.first][e.second+1] == 0 && t[e.first+1][e.second+1] == 0) {
            t[e.first][e.second] = 8;
            t[e.first+1][e.second] = 8;
            t[e.first][e.second+1] = 8;
            t[e.first+1][e.second+1] = 8;
            f.push_back(e);
        }
        cout << "Пуля добавлена" << endl;
        return;
    }
}
cout << "Все места заняты" << endl;

```


5] Вариант N3
← номер макга

```
void show() {
    for (int i = 0; i < n; i++) {
        for (int j = 0; j < n; j++) cout << "i: " << i << "j: " << j << " ";
        cout << endl;
    }
}
```

```
void change-distance() {
    double d; cin >> d; d_ = d / 100 * 8;
    d = d_; cout << "Перемещение за одно генд бие" << endl;
}
```

```
void run (int direct) { // 0 - направо, 1 - влево, 2 - вверх, 3 - вниз
    for (pair<int, int> delta;
        if (direct == 0) delta = {0, 0};
        if (direct == 1) delta = {0, -1};
        if (direct == 2) delta = {1, 0};
        if (direct == 3) delta = {-1, 0};
        for (auto v: f) {
            auto new_v = {v.first + delta.first, v.second + delta.second};
            if (new_v.first == 0) continue;
            if (! (new_v.first > 0 & new_v.second <= n - 1))
                continue;
            if (! (new_v.first <= n - 1 & new_v.second > 0))
                continue;
        }
    }
```


Bapant Vh

Auct → 6

```

    cv.second >
    t cv.first - 13 == 0;
    t cv.first > cv.second == 0;
    t cv.first > t.second - 13 == 0;
    t cv.first > cv.second + 13 == 0;
    t new.v.first - 13 < new.v.second == 0;
    t new.v.first > new.v.second == 0;
    t new.v.first > new.v.second - 13 == 0;
    t new.v.first > new.v.second + 13 == 0;
    v = new v;
    erase(t.find(v));
    t.p

```

v = new v;

```

} cout << "Puzpu egbungry";
int main() {

```

while (true) {

string s; cin >> s;

if (s == "add") add();

if (s == "show") show();

if (s == "distance") change-distance();

if (s == "right") run(0);

if (s == "left") run(1);

if (s == "up") run(2);

if (s == "down") run(3);

}

}