

# Выделение образа черно-белого изображения

Автор: Васильев К. С. РЛ6-21 МГТУ им. Н. Э. Баумана.

Основные действия:

1) Создание хранилища данных (я использовал лист массивов(пара была добавлена для хранения цвета или других данных, но дальше не было использовано))

```
list<pair<bool*,int> > pikcha;
list<pair<bool*,int> > pikchavivod;
int i;
int n;
cout<<"Введите количество строк"<<endl;
cin>>i;
cout<<"Введите количество столбцов"<<endl;
cin>>n;
for (int j = 0; j < i+2; ++j) { //генерим строки
    pikcha.push_back(make_pair(new bool [n+2],1) );
    pikchavivod.push_back(make_pair(new bool [n+2],1) );
}
```

2) Обрамление данных в рамку (для эрозии (но не было использовано, так как Эрозия была реализована вычитанием исходной картинки, а не накладывании маски (если необходимо будет реализовать то будет не сложно исправить)))

```
for (auto iter = pikcha.begin(); iter !=pikcha.end(); iter++) {
    iter->first[0]=1;
    iter->first[n+1]=1;
}
auto iter=pikcha.end();
iter--;
for (int j = 0; j < n+2; ++j) {
    pikcha.begin()->first[j]=1;
    iter->first[j]=1;
}
```

3) Считывание исходной картинки

```
ifstream fin("C:\\tyt_tekst\\lenin.txt");
auto peremen=++pikchavivod.begin();
for (auto iter = ++pikcha.begin(); iter != --pikcha.end(); ++iter,++peremen) {

    for (int k = 1; k < n+2; ++k) {
        ch = fin.get();
        iter->first[k]= ch-48;
```

```

    peremen->first[k]= ch-48;
}
}

```

## 4) Делотация

```

auto peremeniter=pikchavivod.begin();
peremeniter++;

for (auto iter = ++pikcha.begin(); iter != --pikcha.end(); ++iter, ++peremeniter) {

    for (int k = 1; k < n+1; ++k) {
        if (iter->first[k]==1){
            peremeniter->first[k-1]=1;
            peremeniter->first[k+1]=1;
            peremeniter--;
            peremeniter->first[k]=1;
            peremeniter++;
            peremeniter++;
            peremeniter->first[k]=1;
            peremeniter--;
        }
    }
}

```

## 5) Эрозия

```

peremeniter=pikchavivod.begin();
peremeniter++;

for (auto iter = ++pikcha.begin(); iter != --pikcha.end(); ++iter, ++peremeniter) {

    for (int k = 1; k < n+1; ++k) {
        if (iter->first[k]==1){
            peremeniter->first[k]=0;
        }
    }
}

```

## 6) Запись образа исходной картинки в новый файл

```

ofstream fout("C:\\tyt_tekst\\obrazlenina");
for (auto iter = ++pikchavivod.begin(); iter != --pikchavivod.end(); ++iter) {
    for (int k = 1; k < n + 1; ++k) {
        fout << iter->first[k];
        // cout << iter->first[k] << " ";
    }
    fout << "\n ";
    // cout << endl;
}

```

```
}  
fout.close();
```

## Результат работы программы

## Входные данные



Полученный результат

