# Курсовая работа

1. Послание;
2. Границы;
3. Анализ алгоритмов сортировки массивов
4. Формат данных для хранения изображения
5. Чтение заголовка bmp файла

# 1. Послание

**Структура файла-загадки**:

- первый байт - число символов в алфавите
- начиная со второго байта идут символы алфавита в кодировке UniCode (2 байта),
- после алфавита идет текст, в котором один символа занимает один байт - это номер символа из алфавита.

# Курсовая работа

1. Послание;
2. **Границы;**
3. Анализ алгоритмов сортировки массивов
4. Формат данных для хранения изображения
5. Чтение заголовка bmp файла

# 2. Границы

**Границы изображения –** важная часть изображения, по границе можно выделить объект в изображении.

Граница – резкий перепад яркости

Перепад – градиент (производная)

$$\mathbf{G} = \sqrt{\mathbf{G}_x^{\,2} + \mathbf{G}_y^{\,2}} \qquad \mathbf{G}_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ +1 & +2 & +1 \end{bmatrix} * \mathbf{A} \quad \text{and} \quad \mathbf{G}_x = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix} * \mathbf{A}$$

# 2. Границы

Для разработки схемы алгоритмы составить две таблицы: **условия** и **д**

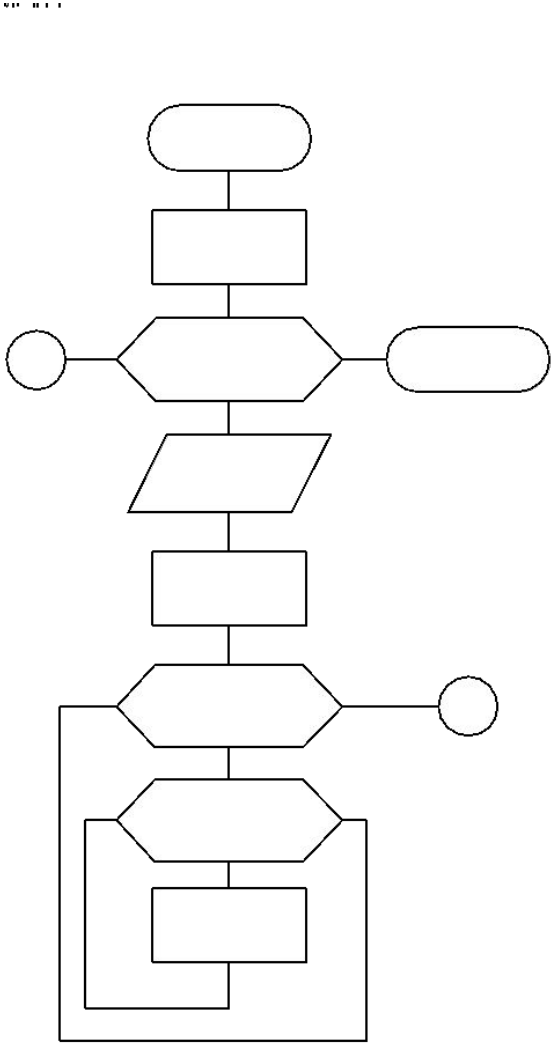
| Код действия | Наименование действия |
|---|---|
| A1 | Создать каталог для результирующих изображений |
| A2 | Получить список всех файлов заданного каталога |
| A3 | Прочитать изображение |
| A4 | Создать изображение для хранения изображения в градациях серого |
| A5 | Преобразовать текущий пиксель из цветного в серый |
| A6 | Записать пиксель в градациях серого |
| A7 | Задание параметров фильтрации |
| A8 | Создать изображение для хранения градиента |
| A9 | Обнулить текущее значение градиента |
| A10 | Рассчитать значение градиента по X и Y для текущего пикселя |
| A11 | Рассчитать значение градиента для текущего пикселя |
| A12 | Записать предельное значение интенсивности |
| A13 | Записать градиент в изображение |

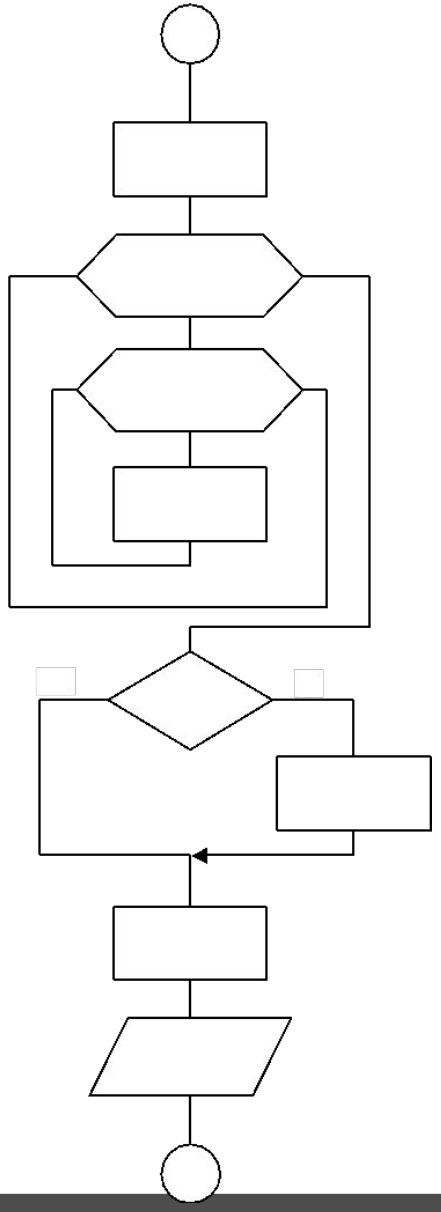
# 2. Границы

| Код действия | Наименование действия |
|---|---|
| A14 | Сохранить изображение с градиентом в файл |

| Код условия | Наименование условия |
|---|---|
| C1 | Выход за границы интенсивности пикселя |

# 2. Границы





# Курсовая работа

1. Послание;
2. Границы;
3. **Анализ алгоритмов сортировки массивов**
4. Формат данных для хранения изображения
5. Чтение заголовка bmp файла
6. Чтение заголовка bmp файла

# 3. Анализ алгоритмов сортировки массивов

**Сортировка пузырком:**

```
ЦИКЛ ДЛЯ J=1 ДО N-1 ШАГ 1
   F=0
   ЦИКЛ ДЛЯ I=1 ДО N-J ШАГ 1
        ЕСЛИ A[I] > A[I+1] ТО ОБМЕН A[I],A[I+1]:F=1
   СЛЕДУЮЩЕЕ I
   ЕСЛИ F=0 ТО ВЫХОД ИЗ ЦИКЛА
 СЛЕДУЮЩЕЕ J
```

# 3. Анализ алгоритмов сортировки массивов

Сложность сортировки пузырьком: $O(n^2)$.

**Наихудший** случай:

- Число сравнений в теле цикла равно $(N-1)\frac{N}{2}$.
- Число сравнений в заголовках циклов равно $(N-1)\frac{N}{2}$ .
- Суммарное число сравнений равно $(N-1)N$ .
- Число присваиваний в заголовках циклов равно $(N-1)\frac{N}{2}$.
- Число обменов равно $(N-1)\frac{N}{2}$.

**Наилучший** случай:

- Число сравнений в теле цикла равно $N-1$.
- Число сравнений в заголовках циклов равно $N$.
- Число присваиваний в заголовках циклов равно $N$
- Число обменов равно 0.

# Курсовая работа

1. Послание;
2. Границы;
3. Анализ алгоритмов сортировки массивов
4. **Формат данных для хранения изображения**
5. Чтение заголовка bmp файла

# 4. Формат данных для хранения изображения

Структура файла:

- заголовок изображения
- изображение

Заголовок изображения (14 байт):

- Ширина – 4 байте
- Высота – 4 байта
- Цвет фона – 6 байт (R, G, B)

Изображение (11N байт):

- Координаты точки (номер строки, номер столбца) – 8 байт
- Цвет – 3 байта (R, G, B)

# Курсовая работа

1. Послание;
2. Границы;
3. Анализ алгоритмов сортировки массивов
4. Формат данных для хранения изображения
5. **Чтение заголовка bmp файла**

# 5. Чтение заголовка bmp файла

**Заголовок файла**

- 0 2 Код 4D42
- 2 4 Размер файла в байтах
- 6 2 0 (Резервное поле)
- 8 2 0 (Резервное поле)
- 10 4 Смещение, с которого начинается само изображение.

**Заголовок BITMAP (Информация об изображении)**

- 14 4 Размер заголовка BITMAP (в байтах) равно 40
- 18 4 Ширина изображения в пикселях
- 22 4 Высота изображения в пикселях
- 26 2 Число плоскостей, должно быть 1
- 28 2 Бит/пиксел: 1, 4, 8 или 24
- 30 4 Тип сжатия
- 34 4 0 или размер сжатого изображения в байтах.
- 38 4 Горизонтальное разрешение, пиксел/м
- 42 4 Вертикальное разрешение, пиксел/м
- 46 4 Количество используемых цветов
- 50 4 Количество "важных" цветов.

# 5. Чтение заголовка bmp файла

**Вариант 1**

- создать запись для заголовка файла
- создать запись для заголовка изображения
- создать запись, которая содержит запись оба типа
- прочитать заголовок файла и изображения одной командой

**Вариант 2**

- прочитать сигнатуру формата
- прочитать смещение
- прочитать размеры изображения
- прочитать тип bmp (бит на пиксель)

# 5. Чтение заголовка bmp файла

## Вариант 1

```
program TestBMP;
type
  TbmFileHeader = record
    Typf : word;
    Size : longword;
    Res1 : word;
    Res2 : word;
    OfBm : longword;
  end;
  TbmInfoHeader = record
    Size : longword;
    Widt : longword;
    Heig : longword;
    Plan : word;
    BitC : word;
    Comp : longword;
    SizI : longword;
    XppM : longword;
    YppM : longword;
    NCoL : longword;
    NCoI : longword;
  end;
```

# 5. Чтение заголовка bmp файла

```
  TbmHeader = record
    f : TbmFileHeader;
    i : TbmInfoHeader;
  end;
var
  fBMP : file;
  bmHeader : TbmHeader;
begin
  assign(fBMP, 'pict.bmp');
  reset(fBMP);
  read(fBMP,bmHeader);
  //читайте палитру, если есть
  //читайте изображение
  close(fBMP);

  writeln('Сигнатура ',bmHeader.f.Typf);
  writeln('Начало изображения ',bmHeader.f.OfBm);
  writeln('Ширина ',bmHeader.i.Widt);
  writeln('Высота ',bmHeader.i.Heig);
  writeln('Бит на пиксель ',bmHeader.i.BitC);
  writeln('Число пикселей ',bmHeader.i.Size);
end.
```

# 5. Чтение заголовка bmp файла

## Вариант 2

```
program TestBMP;
var
  fBMP : file;
  Typf : word;
  OfBm : longword;
  Widt : longword;
  Heig : longword;
  BitC : word;
begin
  assign(fBMP, 'pict.bmp');
  reset(fBMP);
  read(fBMP,Typf);
  seek(fBMP,10);  read(fBMP,OfBm);
  seek(fBMP,18);  read(fBMP,Widt);
  seek(fBMP,22);  read(fBMP,Heig);
  seek(fBMP,28);
  read(fBMP,BitC);
close(fBMP);
  writeln('Сигнатура ',Typf);
  writeln('Начало изображения ', OfBm);
  writeln('Ширина ', Widt);
  writeln('Высота ', Heig);
  writeln('Бит на пиксель ', BitC);
end.
```