

Программирование на языках высокого уровня

Лаб. работа 11. Строки и текстовые файлы

Общие сведения

На оценку «3» - задачи сложности A и B.

На оценку «4» - задачи сложности C

На оценку «5» - задачи сложности D

Требуется написать программу, выполняющую чтение текстового файла и на его основе создание html файла с форматированием, соответствующим заданию. Во всех случаях в html-документе должно быть сохранено разделение на строки, в качестве заголовка должна выступать первая строка файла.

Задача А

- Выделить ***полужирным курсивом*** строки, содержащие букву «А»
- Первую строку отметить тегами <H1>
</H1>

Задача В

- Выделить ***полужирным курсивом*** строки, содержащие хотя бы два одинаковых символа подряд
- Первую строку отметить тегами <H1>
</H1>

Задача С

- Выделить ***полужирным курсивом*** слова, содержащие букву «А», заменить в них её на символ «*»
- Первую строку отметить тегами <H1>
</H1>

Задача D

- Удалить из строки слова, содержащие букву «А», заменить их на слово «NOP», выделить его **жирным курсивом**.
- Первую строку отметить тегами <H1> </H1>

Подсказка

- Формат HTML-документа:

```
<HTML>
```

```
  <HEAD>
```

```
    <TITLE>Заголовок</TITLE>
```

```
  </HEAD>
```

```
  <BODY>
```

```
    <H1>Строка 1<BR></H1>
```

```
    Строка 2<BR>
```

```
    Строка 3<BR>
```

```
  </BODY>
```

```
</HTML>
```

Подсказка

- Теги

`<i>Текст</i>` – *Текст* будет выделен курсивом

`Текст` - **Текст** будет выделен полужирным

`<i>Текст</i>` - ***Текст*** будет выделен полужирным курсивом

`<H1>Текст</H1>` - **Текст** как заголовок (крупный шрифт)

`
` - перенос строки

Подсказка

```
FILE *f; //Файловая переменная
```

```
f = fopen ("ИмяФайла", "rt"); //открытие  
файла для чтения
```

или

```
f = fopen ("ИмяФайла", "wt"); //открытие  
файла для записи
```

```
char s[255]; //строка с максимальной длиной  
255 символов. Строка всегда заканчивается  
символом с кодом 0 – записать его можно как  
'\0'.
```

Подсказка

```
FILE *f; //Файловая переменная
```

```
char s[255]; //строка с максимальной длиной  
255 символов. Строка всегда заканчивается  
символом с кодом 0 – записать его можно как  
'\0'.
```

```
f = fopen ("ИмяФайла", "rt"); //открытие  
файла для чтения
```

или

```
f = fopen ("ИмяФайла", "wt"); //открытие  
файла для записи
```

Подсказка

`fgets (s, 255, f) //прочитать из файла f`
строку максимальной длиной 255 символов

`fputs (s, f) //записать строку s в файл f`

`while (!feof(f)) //пока не конец файла`
`{`
`...`
`}`

Программирование на языках высокого уровня

Лаб. работа 12. Структуры и бинарные
файлы

Общее задание

На оценку «3» - задачи сложности A и B.

На оценку «4» - задачи сложности C

На оценку «5» - задачи сложности D

Требуется написать две программы.

Первая программа: спрашивает у пользователя количество фигур и характеристики каждой фигуры, после чего сохраняет их в файл.

Вторая программа загружает из файла сохраненные туда фигуры и отрисовывает их

Данные фигур в обеих программах хранить в виде структур. Сохранять структуры в файл в бинарном виде. В файле сначала должно храниться число N - количество структур, а затем – сами структуры

Задача А

Фигура – окружность. Сохранять следующие характеристики:

Координаты X , Y

Радиус R

Задача В

Фигура – окружность. Сохранять следующие характеристики:

Координаты X , Y

Цвет R , G , B

Толщина контура W

Радиус R

Задача С

Фигура – четырехлучевая звезда:

Координаты центра X , Y

Цвет R , G , B

Толщина контура W

Радиус R

Задача D

Фигура – надпись

Координаты X, Y

Текст надписи

Опционально: цвет R, G, B , размер
шрифта F

Подсказка

Пример структуры – точка Point

```
struct Point
{
    int X, Y; //Поля структуры
};
```

```
int main()
{
    Point p1;
    ...
    p1.x = 10; //Использование
    p1.y = 15;
    ...
}
```

Подсказка

Бинарные файлы

- Не читаются в «Блокноте»
- Компактнее текстовых
- Хранят данные в том же виде, в каком они хранятся в памяти

```
FILE *f; //Файловая переменная
f = fopen ("ИмяФайла", "rb"); //открытие файла для чтения
или
f = fopen ("ИмяФайла", "wb"); //открытие файла для записи
```

Запись в бинарный файл

```
int X;
Point px;
...
fwrite (&X, sizeof(int), 1, f);           //Запись целого
числа в файл
fwrite (&px, sizeof(Point), 1, f);        //Запись структуры
Point в файл
```

Подсказка

Чтение из бинарного файла

```
int X;  
Point px;  
...  
fread (&X, sizeof(int), 1, f);           //Запись целого  
числа в файл  
fread (&px, sizeof(Point), 1, f);       //Запись структуры  
Point в файл
```