

ТЕОРИЯ ВЫЧИСЛИТЕЛЬНЫХ ПРОЦЕССОВ И СТРУКТУР.

Лабораторная работа 3.

Обработка текста в языках программирования.

Составить программу, выполняющую по порядку следующие предложенные задания, отображая результаты каждого этапа на экране дисплея.

Считать текст из файла и сохранить его в массиве строк. Используя данное представление текста выполнить следующие действия:

- вывод текста на экран, причем верхняя строка экрана должна быть оформлена в качестве меню работы с текстом;
- реализовать "прокрутку" текста в заданном окне;
- вывести заданный фрагмент текста с i -й по j -ю строки;
- возможность редактирования любой строки текста;
- вывести на экран фрагмент исходного текста, используя прямое обращение к видеопамяти;
- вывести на экран фрагмент исходного текста с использованием прерываний;
- выполнить индивидуальные задания в зависимости от варианта.

Варианты заданий:

1. Реализовать функцию подсчета слов в тексте.
2. Реализовать функцию перестановки местами двух строк исходного текста.
3. Реализовать функцию подсчета количества гласных букв в заданной строке и выделение их цветом.
4. Реализовать функцию добавления строки в любое место исходного текста (при нажатии клавиши INS).
5. Реализовать функцию удаления заданного слова из строки и выделения цветом предыдущего и последующего слова.
6. Реализовать функцию удаления строки из исходного текста (при нажатии клавиши DEL).
7. Реализовать функцию поиска заданного слова в тексте и выделение его цветом.
8. Реализовать функцию подсчета числа слов в заданной строке.
9. Реализовать функцию добавления слова в указанное место строки и выделение этого слова цветом.
10. Реализовать функцию удаления первого и последнего слова в заданной строке.
11. Реализовать функцию поиска по всему исходному тексту заданной комбинации символов и выделение цветом.
12. Реализовать функцию сортировки слов в тексте по возрастанию.
13. Реализовать функцию сортировки букв по возрастанию.
14. Реализовать функцию нахождения слов, имеющих заданное окончание.

15. Реализовать функцию подсчёта числа вхождений каждого слова в текст.

Необходимые рекомендации к выполнению.

Составлять программу желательно на языке программирования Delphi.

Последовательность действий следующая:

1) С помощью Menu Designer добавьте в форму полосу меню. Эта полоска имеет три выпадающих меню: меню File с единственной командой Exit; меню Options с командами Font, Color, Left, Center, Hight (между Color и Left установите разделитель, для Left, Center, Hight назначьте горячие клавиши); меню Help с элементом About. Чтобы поместить разделитель, вместо текста команды вставьте просто дефис.

2) В форму поместите компонент RichEdit и две пиктограммы диалогов FontDialog и ColorDialog.

3) Для отклика на команды меню вы должны определить метод для события OnClick каждого элемента меню. Код метода TForm1.Font1Click:

```
procedure TForm1.Font1Click (Sender : TObject) ;
```

```
begin
```

```
    FontDialog1.Font := RichEdit1.Font ;
```

```
    FontDialog1.Execute;
```

```
    RichEdit1.Font := FontDialog1.Font ;
```

```
end;
```

4) Код метода TForm1.Color1Click аналогичен предыдущему.

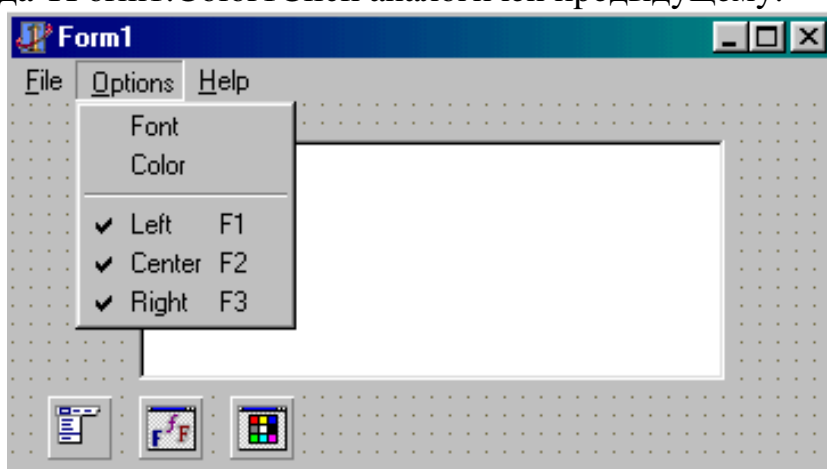


Рис.18 . Шаблон приложения

5) Код метода TForm1.Left1Click:

```
procedure TForm1.Left1Click (Sender : TObject) ;
```

```
begin
```

```
    RichEdit1.Paragraph.Alignment := taLeftJustify ;
```

```
    Left1.Checked := True ;
```

```
    Center1.Checked := False ;
```

```
    Right1.Checked := False ;
```

```
end;
```

Чтобы поставить галочку в ряде выбираемых опций, установите свойство Checked элемента меню в окне Object Inspector в True.

6) Код методов для элементов Center и Right аналогичен предыдущему.

7) На форме расположить две панели, две кнопки и компонент RichEdit. Первая панель содержит два поля редактирования, а вторая два чекбокса. Также необходимо построить выпадающее меню File, Buttons, View, Pulldowns. Команды меню File – Open , SaveAs. Команды меню Buttons содержит изменяемый текст (с ‘Enable First’ на ‘Disable First’). Команды меню View – Edit Boxes, Check Boxes. Команды Pulldowns – Remove File, Disable Buttons, Disable View. Поместите в форму пиктограммы необходимых диалогов.

8) Код методов, которые загружают и сохраняют файлы:

```
procedure TForm1.Open1Click (Sender : TObject) ;
begin
  if OpenFileDialog1.Execute then
    RichEdit1.Lines.LoadFromFile (OpenDialog1.FileName) ;
end;
procedure TForm1.SaveAs1Click (Sender : TObject) ;
begin
  if SaveDialog1.Execute then
    RichEdit1.Lines.SaveToFile (SaveDialog1.FileName) ;
end;
```

9) Компоненты внутри панелей в действительности не используются. Однако вам необходимо воспользоваться двумя кнопками, чтобы скрыть или отобразить каждую из двух панелей вместе с управляющими элементами, которые в них содержатся. Те же действия можно выполнить с помощью двух команд меню : View/ Edit Boxes и View/ Check Boxes. Когда вы выбираете одну из этих команд меню или нажимаете одну из кнопок, происходит три разных действия. Во-первых, отображается или скрывается панель. Во-вторых, текст кнопки изменяется с Hide на Show, и наоборот. В-третьих, рядом с соответствующим элементом меню появляется или исчезает галочка. Ниже приведен код одного из двух методов, который связан с событиями щелчка как команды меню, так и кнопки:

```
procedure TForm1.ViewEdit1Click (Sender : TObject) ;
begin
  Panel1.Visible := not Panel1.Visible ;
  ViewEdit1.Checked := not ViewEdit1.Checked ;
  if Panel1.Visible then
    Button1.Caption := ‘Hide’ ;
  else
    Button1.Caption := ‘Show’ ;
end;
```

10) Команды меню Buttons применяют другой подход. Для показа текущего состояния они используют не галочку, а изменение текста. Кроме того, они разрешают или запрещают соответствующую команду View и кнопку.

```
procedure TForm1.ButtonsFirst1Click (Sender : TObject) ;  
begin  
  if Buttons1.Enabled then  
  begin  
    Buttons1.Enabled := False;  
    ViewEdit1.Enabled := False ;  
    ButtonsFirst1.Caption := 'Enable &First' ;  
  end  
  else  
  begin  
    Buttons1.Enabled := True  
    ViewEdit1.Enabled := True ;  
    ButtonsFirst1.Caption := 'Disable &First' ;  
  end  
end;  

```

Индивидуальные задания нужно реализовать самостоятельно путем добавления строк программного кода к приведённому коду.