Лабораторная работа № 15 гр.252005 Винничка Кирилла(2 вариант)

unit Unit1;

interface

uses

Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,

Dialogs, StdCtrls,Unit2;

var

Form1: TForm1;

Stack: TResh;

implementation

{$R \*.dfm}

procedure TForm1.FormCreate(Sender: TObject);

Создать класс (в Unit2), реализующий стандартные методы работы со сте-

ком Adds, Reads, Print, ReadAfter, AddAfter, Poisk,

PoiskAfter, SortBublInf, SortBublAfter. Написать программу

(Unit1), иллюстрирующую работу всех методов работы со стеком. Результат

формирования и преобразования стека показывать в компонентах TListBox.

После этого на базе стандартного родительского класса написать свой класс,

реализующий метод решения своего варианта. Написать обработчик события,

реализующий вызов метода решения своего варианта.

2. Создать стек из случайных целых чисел и удалить из него записи с чет-

ными числами.

begin

Stack:=Tresh.create;

end;

procedure TForm1.Button1Click(Sender: TObject);

var i:integer;

begin

if stack.sp1<>nil then

for i:=1 to n do stack.Reads(b);

ListBox1.Clear;

ListBox2.Clear;

Randomize;

n:=7;

for i:=1 to n do

stack.Adds(random(20)+1);

stack.print(ListBox1);

end;

procedure TForm1.Button3Click(Sender: TObject);

begin

ListBox1.Clear;

n:=n+1;

stack.Adds(random(20)+1);

stack.print(ListBox1);

end;

procedure TForm1.Button5Click(Sender: TObject);

begin

n:=n-1;

Stack.Reads(b);

Listbox1.Items.Delete(0);

end;

procedure TForm1.Button2Click(Sender: TObject);

begin

Listbox1.Clear;

stack.SortBublInf;

stack.Print(ListBox1);

end;

procedure TForm1.Button4Click(Sender: TObject);

begin

ListBox2.clear;

Stack.resh;

Stack.Print(ListBox2);

end; end.

unit Unit2;

interface

uses

Windows, Messages, SysUtils, Variants, Classes, Graphics, Controls, Forms,

Dialogs, StdCtrls;

Type Tsel=^sel;

Sel=record

inf:integer;

A:Tsel;

end;

TStack=class(tobject)

sp,sp1:Tsel;

procedure Adds(Inf:integer);

procedure Reads(var Inf:integer);

procedure Print(ListBox:TListBox);

procedure AddAfter(spi:Tsel; Inf:integer);

procedure ReadAfter(spi:tsel; var Inf:integer);

function Poisk(Inf:integer):Tsel;

Procedure SortBublInf;

Procedure SortBublAfter;

end;

TResh=class(Tstack)

hp,hp1:Tsel;

constructor create;

procedure resh;

end;

var n,m:word; b:integer;

implementation

constructor TResh.create;

begin

inherited create;

sp1:=nil; hp1:=nil;

end;

procedure Tstack.Adds;

begin

New(sp);

sp^.inf:=inf;

sp^.A:=sp1;

sp1:=sp;

end;

procedure Tstack.Reads;

begin

inf:=sp1^.inf;

sp:=sp1;

sp1:=sp1^.A;

Dispose(sp);

end;

procedure Tstack.Print;

begin

sp:=sp1;

While sp<>nil do begin

ListBox.Items.Add(IntTostr(sp^.inf));

sp:=sp^.A;

end;

end;

procedure Tstack.AddAfter;

begin

New(sp);

Sp^.inf:=inf;

sp^.A:=spi^.A;

spi^.A:=sp;

end;

procedure Tstack.ReadAfter;

begin

sp:=spi^.A;

inf:=sp^.inf;

spi^.A:=sp^.A;

dispose(sp);

end;

Function Tstack.Poisk;

begin

if sp1=Nil then Result:=nil

else begin

sp:=sp1;

while (sp^.inf mod 2=1) and (sp^.a<>nil) do Sp:=sp^.A;

if sp^.inf mod 2=1 then result:=nil else result:=sp;

end;

end;

Procedure Tstack.SortBublInf;

procedure RevInf(spi:Tsel);

var inf:integer;

begin

inf:=sp^.inf;

sp^.inf:=sp^.A^.inf;

sp^.A^.inf:=inf;

end;

var spt:Tsel;

begin

spt:=nil;

repeat

sp:=sp1;

while sp^.A<>spt do begin

if sp^.Inf>sp^.A^.inf then RevInf(sp);

sp:=sp^.a;

end;

spt:=sp;

until sp1^.A=spt;

end;

Procedure TStack.SortBublAfter;

Procedure RevAfter(spi:Tsel);

var sp:tsel;

begin

sp:=spi^.A^.A;

spi^.A^.A:=sp^.A;

sp^.A:=spi^.A;

spi^.A:=sp;

end;

Var spt:Tsel;

begin

spt:=Nil;

Repeat

sp:=sp1;

While sp^.A^.A<>spt do begin

if sp^.A^.Inf mod 2=1

then RevAfter(sp);

sp:=sp^.A;

end;

spt:=sp^.A;

Until sp1^.A^.A=spt;

end;

procedure Tresh.resh;

var spt:Tsel; i:word;

begin

m:=0;

SortBublafter;

sp:=sp1;

While sp^.inf mod 2=0 do begin inc(m); sp:=sp^.a; end;

for i:=1 to m do Reads(b);

end;

end.