

Практична робота №13

Тема: Знаходження максимального та мінімального елементів масиву.

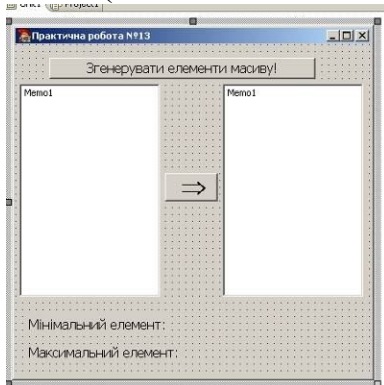
Мета: Ознайомитися із алгоритмом пошуку максимального та мінімального елементів у невідсортованому масиві.

Хід роботи

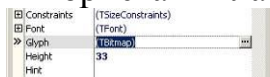
1. Створити у своїй робочій папці каталог PRACT13.
2. Запустити середовище Delphi та створити новий додаток.
3. Виконати File – Save All, вказавши пунктом призначення створений Вами каталог PRACT13, під іменами, які пропонує середовище Delphi.

Умова задачі. В одновимірному масиві, який складається із 15 дійсних випадкових чисел в діапазоні від -100 до 100, визначити мінімальний та максимальний елементи, а також переписати всі елементи вхідного масиву, які розміщені між цими елементами, до вихідного масиву.

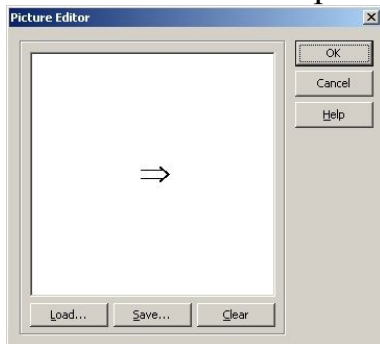
4. Деякі зауваження щодо проектування форми. Доцільно на формі розмістити кнопку Button1 (Згенерувати елементи масиву!), два поля Memo (для виведення елементів вхідного та вихідного масивів), кнопку SpeedButton1 (⇒), та два написи Label (Мінімальний елемент: та Максимальний елемент:)



Щоб розмістити на кнопці SpeedButton малюнок, його потрібно спочатку створити (наприклад, засобами графічного редактора Paint), зберегти у папці Pract13, а потім використати властивість Glyph:



У Picture Editor потрібно завантажити створений Вами малюнок:



5. Деякі зауваження щодо програмного коду для реалізації даного проекту.
 - 5.1. У програмі використовуються 2 глобальні змінні для опису масивів дійсних чисел (вхідного та вихідного):

```
var  
    Form1: TForm1;  
28   a,b: array [1..15] of real;  
30 implementation  
    ($R *.dfm)
```

5.2. У програмі доцільно використати три процедури обробки подій: відкриття форми **procedure TForm1.FormCreate (Sender: TObject);** натискування на кнопку мишки на об'єктах (2 процедури): **procedure TForm1.Button1Click (Sender: TObject);** та **procedure TForm1.SpeedButton1Click (Sender: TObject);**

6. Програмний код для процедури **procedure TForm1.FormCreate (Sender: TObject);**

```
{ $R *.dfm }  
procedure TForm1.FormCreate(Sender: TObject);  
begin  
    memo1.Clear;  
    memo2.Clear;  
    randomize;  
end;
```

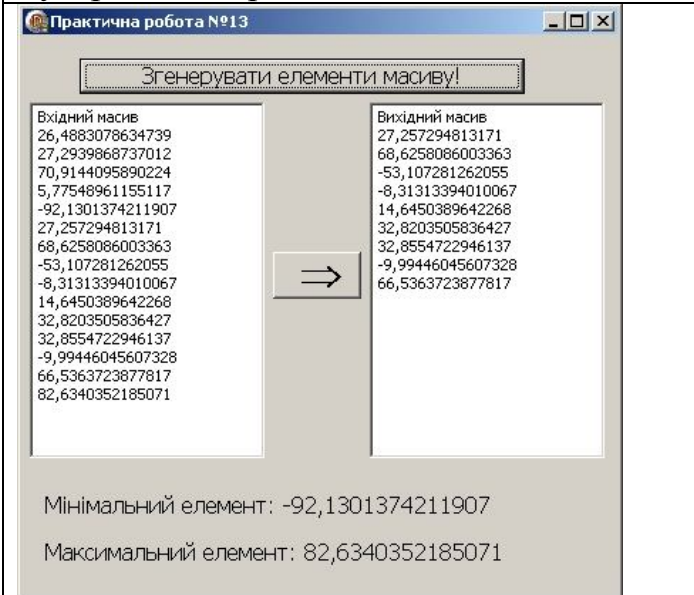
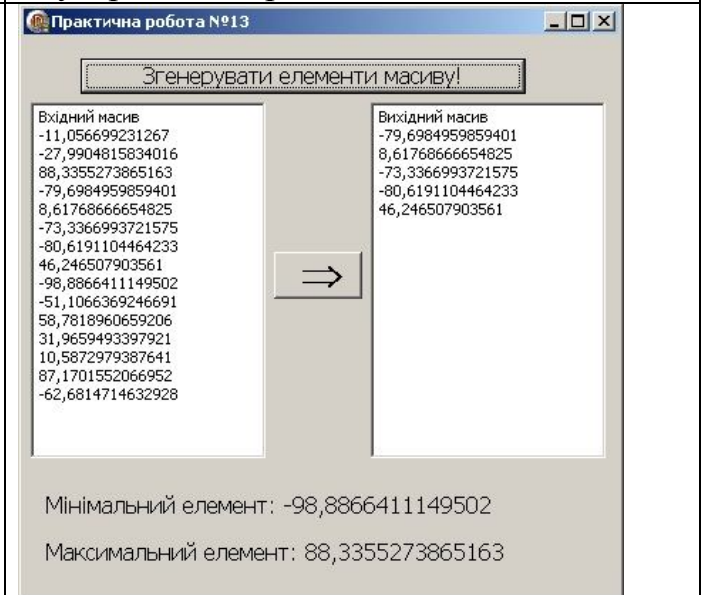
7. Програмний код для процедури **procedure TForm1.Button1Click (Sender: TObject);**

```
procedure TForm1.Button1Click(Sender: TObject);  
var i: integer;  
begin  
    memo1.Clear;  
    memo1.Lines.Add('Вхідний масив');  
    memo2.Clear;  
    memo2.Lines.Add('Вихідний масив');  
    for i:=1 to 15 do //генеравання елементів масиву  
        a[i]:=random*200-100;  
    for i:= 1 to 15 do //виведення елементів масиву  
        memo1.Lines.Add(floattostr(a[i]));  
end;
```

8. Програмний код для процедури **procedure TForm1.SpeedButton1Click (Sender: TObject);**

```
procedure TForm1.SpeedButton1Click(Sender: TObject);  
var i, pmin, pmax, p: integer;  
begin  
    pmin:=1; //задати початкові значення для індекса мінімуму  
    pmax:=1; //задати початкові значення для індекса максимуму  
    for i:=2 to 15 do //перебрати усі елементи масиву  
        begin  
            if a[i]<a[pmin] then //якщо елемент менший від мінімального  
                pmin:=i; //запам'ятати його номер  
            if a[i]>a[pmax] then //якщо елемент більший від максимального  
                pmax:=i; //запам'ятати його номер  
        end;  
    //вивести значення мінімального та максимального елементів  
    label1.Caption := 'Мінімальний елемент: '+floattostr(a[pmin]);  
    label2.Caption := 'Максимальний елемент: '+floattostr(a[pmax]);  
    //сформувати вихідний масив  
    p:=0; //початкова позиція в масиві  
    if pmin<pmax then //якщо раніше зустрівся мінімум  
        for i:=pmin+1 to pmax - 1 do  
            begin  
                inc(p); //зсунути позицію p  
                b[p]:=a[i]; //записати в позицію p  
            end  
        else //інакше - якщо раніше зустрівся максимум  
            for i := pmax+1 to pmin - 1 do  
                begin  
                    inc(p); //зсунути позицію p  
                    b[p]:=a[i]; //записати у позицію p  
                end;  
    for i := 1 to p do //вивести вихідний масив  
        memo2.Lines.Add(floattostr(b[i]));  
end;
```

9. Зберегти отриманий додаток на диску та запусимо його на виконання.
10. Завершити роботу із середовищем Delphi.
11. Із операційної оболонки запустити файл із розширенням exe, який утворився у створеній Вами папці PRACT13. Важливо! Доцільно виконати декілька разів генерацію, щоб пересвідчитися у тому, що створений Вами проект працює як у випадку, коли мінімальний елемент відшукано раніше, ніж максимальний, так і у випадку, коли максимальний елемент передує мінімальному. До звіту вставити два скрін-шоти, наприклад:

Випадок 1. Мінімальний елемент у масиві зустрічається раніше, ніж максимальний	Випадок 2. Максимальний елемент у масиві зустрічається раніше, ніж мінімальний
 <p>Вхідний масив</p> <pre>26,4883078634739 27,2939868737012 70,9144095890224 5,77548961155117 -92,1301374211907 27,257294813171 68,6258086003363 -53,107281262055 -8,31313394010067 14,6450389642268 32,8203505836427 32,8554722946137 -9,99446045607328 66,5363723877817 82,6340352185071</pre> <p>Вихідний масив</p> <pre>27,257294813171 68,6258086003363 -53,107281262055 -8,31313394010067 14,6450389642268 32,8203505836427 32,8554722946137 -9,99446045607328 66,5363723877817</pre> <p>Мінімальний елемент: -92,1301374211907 Максимальний елемент: 82,6340352185071</p>	 <p>Вхідний масив</p> <pre>-11,056699231267 -27,9904815834016 88,3355273865163 -79,6984959859401 8,61768666654825 -73,3366993721575 -80,6191104464233 46,246507903561 -98,8866411149502 -51,1066369246691 58,7818960659206 31,9659493397921 10,5872979387641 87,1701552066952 -62,6814714632928</pre> <p>Вихідний масив</p> <pre>-79,6984959859401 8,61768666654825 -73,3366993721575 -80,6191104464233 46,246507903561</pre> <p>Мінімальний елемент: -98,8866411149502 Максимальний елемент: 88,3355273865163</p>

12. Навести порядок на робочому місці.