

# МАСИВЫ

**Мета:** Ознайомитися із однією із найбільш розповсюджених структур даних – масивами; навчитися вводити масиви, розраховувати для них базові характеристики, відшукувати елементи у масиві.

**Масивом** називається впорядкована індексована сукупність однотипних елементів, які мають спільне ім'я. Елементами масиву можуть бути дані різних типів, у тому числі, й структуровані.

Кожен елемент одновимірного масиву однозначно визначається іменем та індексом (для багатовимірних – індексами).

Масиви зручно використовувати там, де потрібно працювати із великою кількістю однотипних даних.

При цьому передбачається багаторазова їх обробка.

Масиви зручно використовувати там, де потрібно працювати із великою кількістю однотипних даних.

При цьому передбачається багаторазова їх обробка.

**Одновимірний (лінійний) масив** – це масив, у якого в описі задано лише один індекс.

Одновимірні масиви часто називають векторами, так як вони являють собою скінченну послідовність пронумерованих елементів.

Загальний вигляд опису лінійного  
(одновимірнього) масиву:

**<Ім'я масиву>:array[<нижня межа>..  
<верхня межа>] of<тип елементів>;**

Var

**B: array [1..5] of integer;**

**C: array [0..9] of boolean;**

**Масиви** відносяться до структур прямого доступу. При цьому вміст масиву зберігається в неперервній області пам'яті. Це значить, що можна напряду (не перебираючи всі попередні) звернутися до потрібного елементу масиву. Для цього необхідно вказати ім'я масиву та індекс (індекси), розміщені в квадратних дужках:



## Розміщення масиву А цілих чисел у пам'яті:

A[1]	A[2]	A[3]	A[4]	A[5]
Базова адреса	Базова адреса + 4 байти	Базова адреса + 8 байт	Базова адреса + 12 байт	Базова адреса + 16 байт

Всього під масив, наведений у таблиці, в пам'яті буде відведено 20 байт – оголошений масив А цілих чисел типу `integer`, кожному елементу (комірці) масиву буде виділено по 4 байти пам'яті.

Всі ці елементи відповідають звичайним змінним цього типу.

## Введення даних у масив. Відображення його вмісту.

При вивченні мови програмування Паскаль для введення даних до масиву, використовувався цикл For:

```
var a: array [1..10] of integer;
```

```
  i: integer;
```

```
  begin
```

```
    for i:=1 to 10 do
```

```
      begin
```

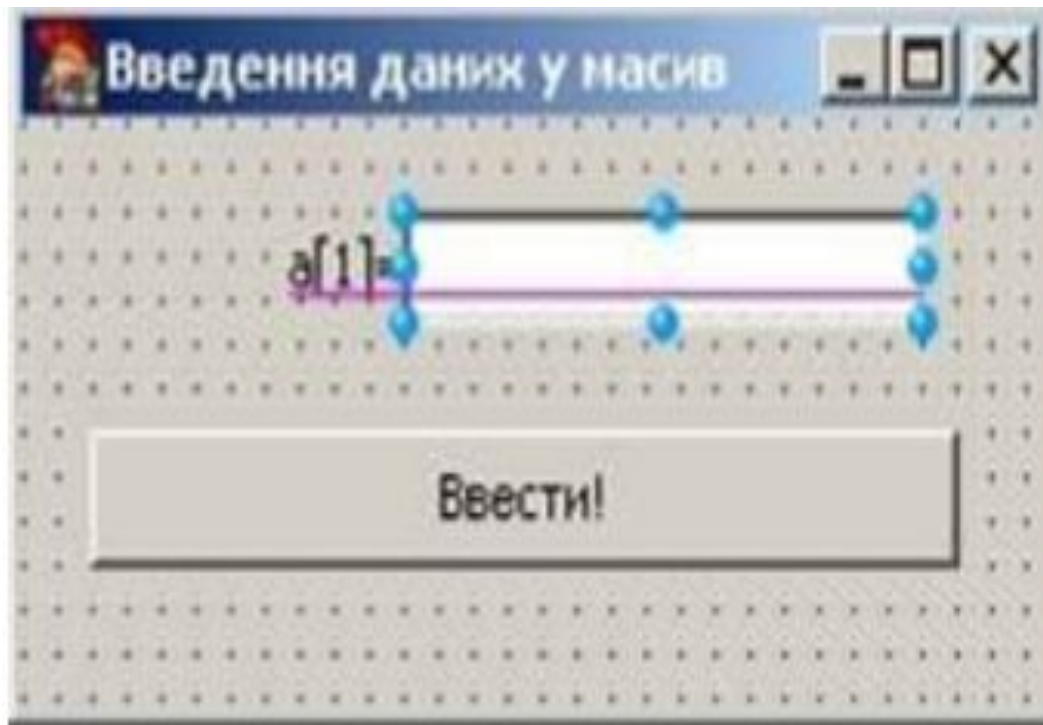
```
        write ('a[', i, ']=');
```

```
        readln (a[i]);
```

```
      end;
```

```
    end;
```

**Завдання1.** Розробити програму, яка реалізує заповнення на формі масиву цілих чисел, який містить 10 елементів (Massiv01).





**var**

```
Form1: TForm1;  
a: array [1..10] of integer;  
i: integer=1;
```

**implementation**

```
($R *.dfm)
```

**procedure** TForm1.Button1Click(Sender: TObject);

**begin**

```
  a[i]:=strtoint(edit1.text);
```

```
  label1.caption:='a[' + inttostr (i+1) + ']=';
```

```
  i:=i+1;
```

```
  edit1.clear;
```

```
  edit1.setfocus;
```

```
  if i=11 then
```

```
  begin
```

```
    label1.visible:=false;
```

```
    edit1.enabled:=false;
```

```
    button1.enabled:=false;
```

```
  end;
```

```
end;
```

При вивченні мови програмування Паскаль для виведення

елементів масиву використовувався цикл For:

```
var a: array [1..10] of integer;
```

```
i: integer;
```

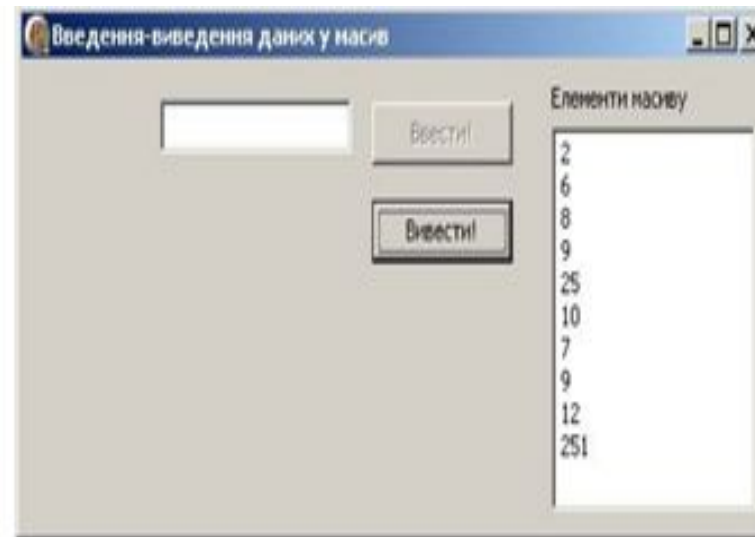
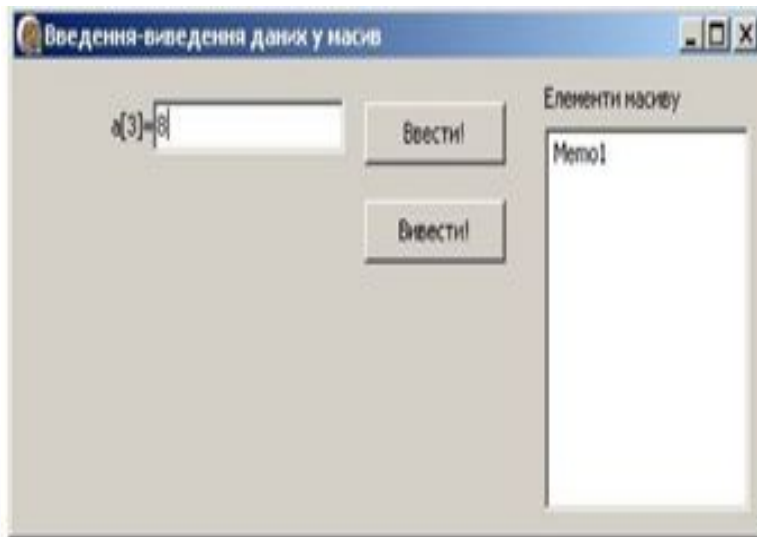
```
begin
```

```
for i:=1 to 10 do
```

```
write ('a[', i, ']=', a[i]);
```

```
end;
```

**Завдання 2.** Доповнити попередній проект таким чином, щоб по завершенні введення елементи масиву виводились на форму (Massiv02).





## Фрагмент программного коду:

**var**

Form1: TForm1;

a: array [1..10] of integer;

i: integer=1;

k: integer;

**implementation**

(\$R \*.dfm)

```
procedure TForm1.Button1Click(Sender: TObject);
begin
  a[i]:=strtoint(edit1.text);
  label1.caption:='a[' + inttostr (i+1) + ']=';
  i:=i+1;
  edit1.clear;
  edit1.setfocus;
  if i=11 then
  begin
    label1.visible:=ffalse;
    edit1.enabled:=false;
    button1.enabled:=false;
  end;
end;
```

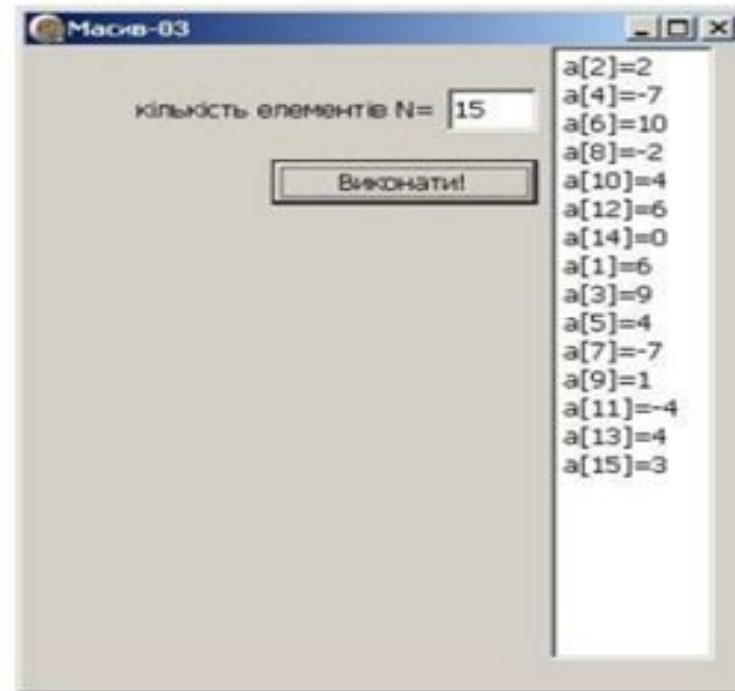
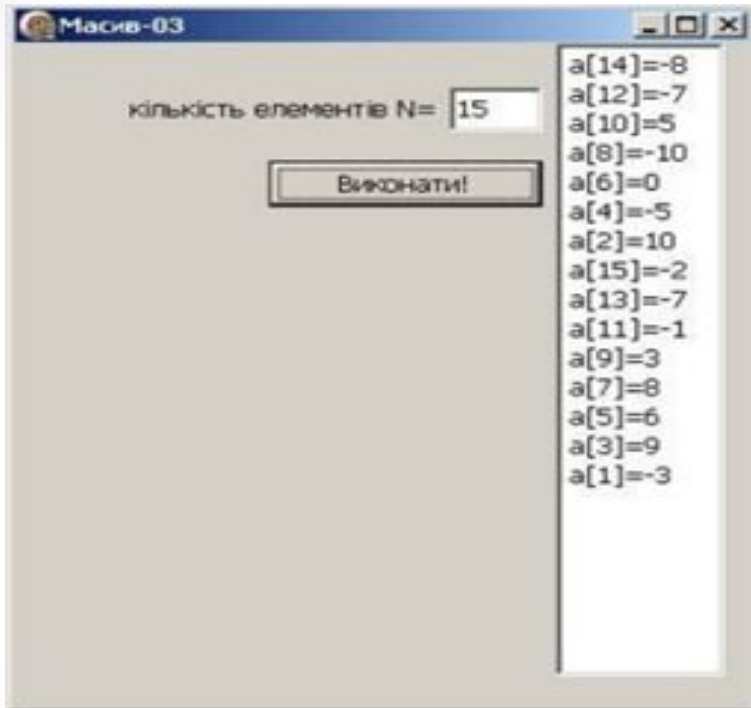
```
procedure TForm1.Button2Click(Sender: TObject);  
begin
```

---

```
    for k:=1 to 10 do  
        memo1.Lines.Add(inttostr(a[k]));  
end;
```

**Завдання 3.** Дано масив із  $N$  ( $N \leq 50$ ) цілих чисел. Його потрібно заповнити випадковими числами у діапазоні від -10 до 10. Вивести спочатку всі елементи, розміщені на парних позиціях, а потім – всі елементи, розміщені на непарних позиціях. При цьому якщо перший елемент більший від останнього елемента, то виведення повинне здійснюватися від початку до кінця масиву, у протилежному випадку – від кінця до початку.

# Роботу із статичним масивом демонструє наступна програма(Massiv03):



У програмі доцільно використати дві процедури – відкриття форми та дія по натискуванні на кнопку **Виконати**.

```
type
  TForm1 = class(TForm)
    Label1: TLabel;
    Edit1: TEdit;
    Button1: TButton;
    Memo1: TMemo;
    procedure Button1Click(Sender: TObject);
    procedure FormCreate(Sender: TObject);
  private
    { Private declarations }
  public
    { Public declarations }
  end;
```

```
procedure TForm1.FormCreate(Sender: TObject);  
begin  
  randomize; |  
end;
```

```

procedure TForm1.Button1Click(Sender: TObject);
begin
    mem1.Clear;
    n:=strtoint (edit1.Text); //ввести кількість елементів
    for i:=1 to n do
        a[i]:=random(21)-10; //заповнити випадковими числами
    if a[1]>a[n] then //якщо перший елемент більший від останнього,
        begin //вивести від 1 до останнього
            for i:=1 to n do
                if i mod 2 =0 then //перевірка на парність
                    mem1.lines.Add('a['+inttostr(i)+']='+inttostr(a[i]));
            for i:=1 to n do
                if i mod 2 <>0 then //перевірка на непарність
                    mem1.lines.add('a['+inttostr(i)+']='+inttostr(a[i]));
        end
    else //якщо перший елемент не більший від останнього
        begin //виведення від останнього до першого
            for i:=n downto 1 do
                if i mod 2 = 0 then
                    mem1.lines.add('a['+inttostr(i)+']='+inttostr(a[i]));
            for i:=n downto 1 do
                if i mod 2 <> 0 then
                    mem1.lines.add('a['+inttostr(i)+']='+inttostr(a[i]));
        end;
end;
end;

```



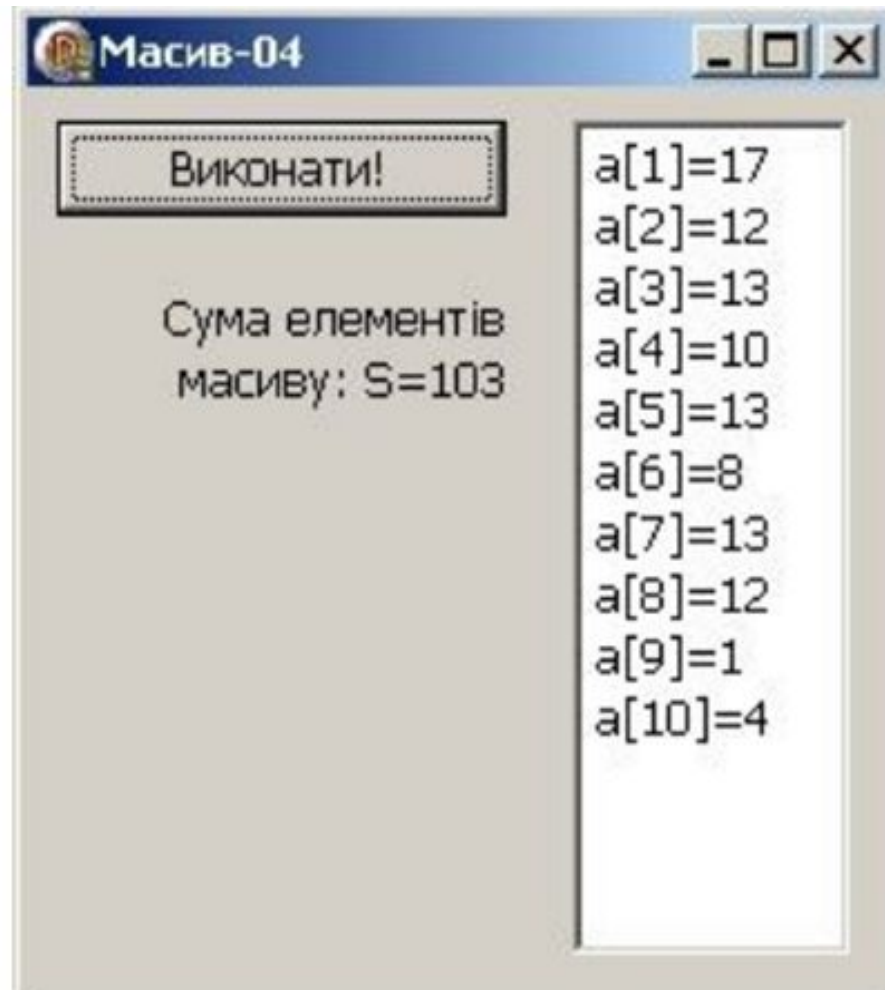
**Процес накопичення суми елементів масиву досить простий і практично нічим не відрізняється від додавання значень деякої числової послідовності.**

**Змінній, у якій зберігається значення суми, присвоюється значення, рівне 0, потім у циклі послідовно додаються елементи масиву.**

Як правило, при роботі з масивом використовується **цикл For** – це пов'язане із тим, що в даній структурі **кількість ітерацій наперед відома.**

**Тип змінної для збереження суми повинен бути таким же, як і тип елементів масиву.**

**Задача 4.** Знайти суму усіх елементів масиву А, який складається із 10 цілих випадкових чисел у діапазоні від 0 до 20 (Massiv04).



У розділі Var опишемо масив та дві змінні типу цілі числа:

**var**

```
Form1: TForm1;
```

```
a: array [1..10] of integer;
```

```
i, s: integer;
```

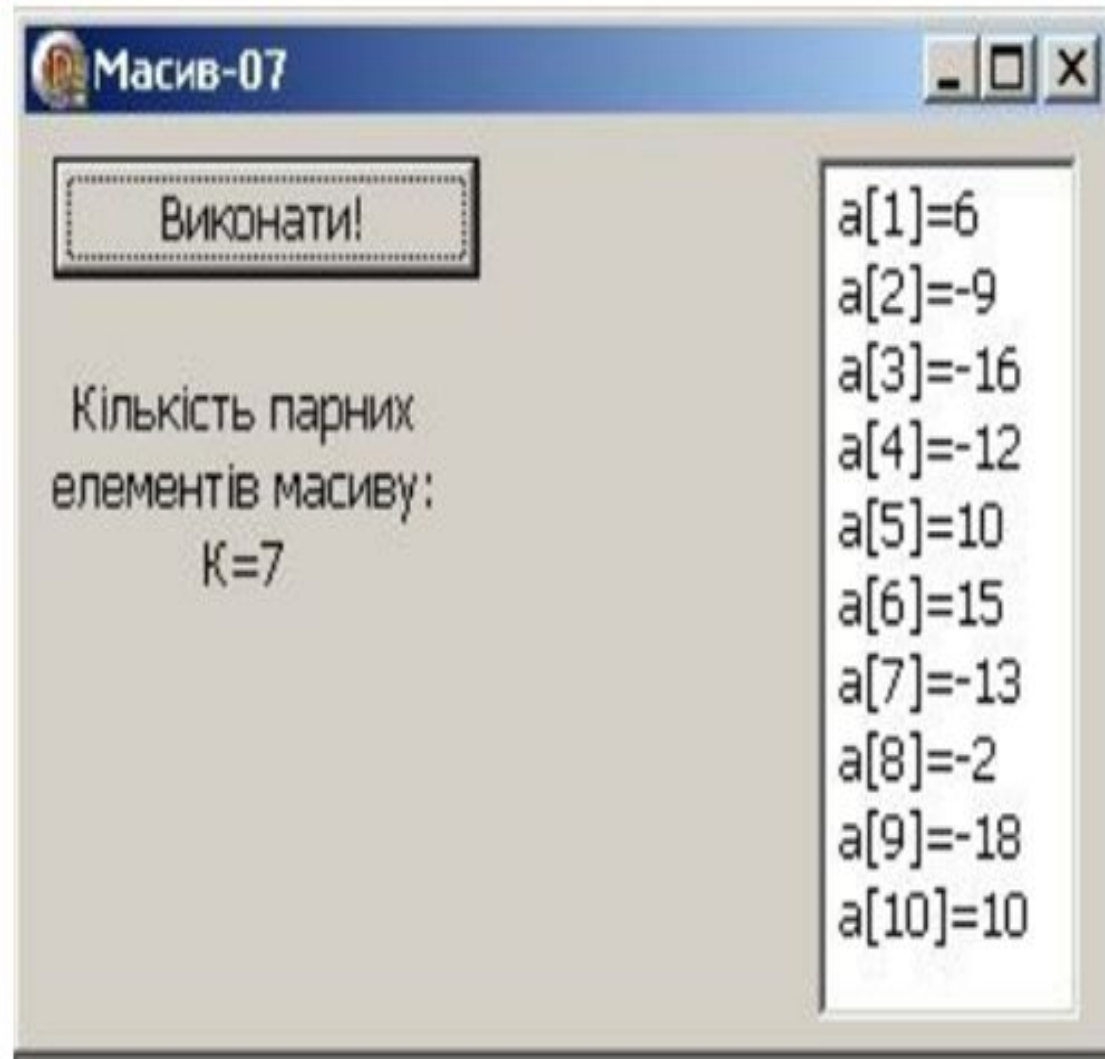
У процедурі щодо відкриття форми доцільно прописати виконання процедури `randomize` – генератора випадкових чисел, очищення вмісту поля Мемо, генерацію елементів масиву та їх виведення на форму:

```
procedure TForm1.FormCreate(Sender: TObject);
begin
  randomize;
  memo1.Clear;
  for i:=1 to 10 do //заповнити масив випадковими числами
    a[i]:=random(20);
  for i:= 1 to 10 do //вивести елементи масиву у полі Мемо
    memo1.lines.add('a['+inttostr(i)+']='+inttostr(a[i]));
end;
```

У процедурі обробника події «Натиснути на кнопку Виконати!» власне й реалізовано розв'язок задачі:

```
procedure TForm1.Button1Click(Sender: TObject);
begin
    s:=0;           //початкова сума
    for i:=1 to 10 do //додати всі елементи масиву
        s:=s+a[i];
    label1.Caption:='Сума елементів'+#13+'масиву: S='+inttostr(s);
                    //вивести значення суми
end;
```

**Задача 5.** Знайти кількість парних елементів масиву А, який складається із 10 цілих випадкових чисел у діапазоні від -20 до 20 (Massiv07).



## Програмний код для розв'язування цієї задачі наступний:

**var**

```
Form1: TForm1;  
a: array [1..10] of integer;  
i, k: integer;
```

**implementation**

```
($R *.dfm)
```

```
procedure TForm1.Button1Click(Sender: TObject);
```

```
begin
```

```
    k:=0;           //початкова кількість  
    for i:=1 to 10 do //перевірити всі елементи масиву  
        if (a[i] mod 2 =0) then k:=k+1; //якщо елемент парний, то  
            //кількість збільшити на 1  
    label1.Caption:='Кількість парних'+#13+  
        'елементів масиву:'+#13+ 'K='+inttostr(k);  
            //вивести значення кількості
```

```
end;
```



```
procedure TForm1.FormCreate(Sender: TObject);
begin
  randomize;
  mem1.Clear;
  for i:=1 to 10 do           //заповнити масив випадковими числами
    a[i]:=random(40)-20;
  for i:= 1 to 10 do         //вивести елементи масиву у полі Memo

    mem1.lines.add('a['+inttostr(i)+']='+inttostr(a[i]));
end;

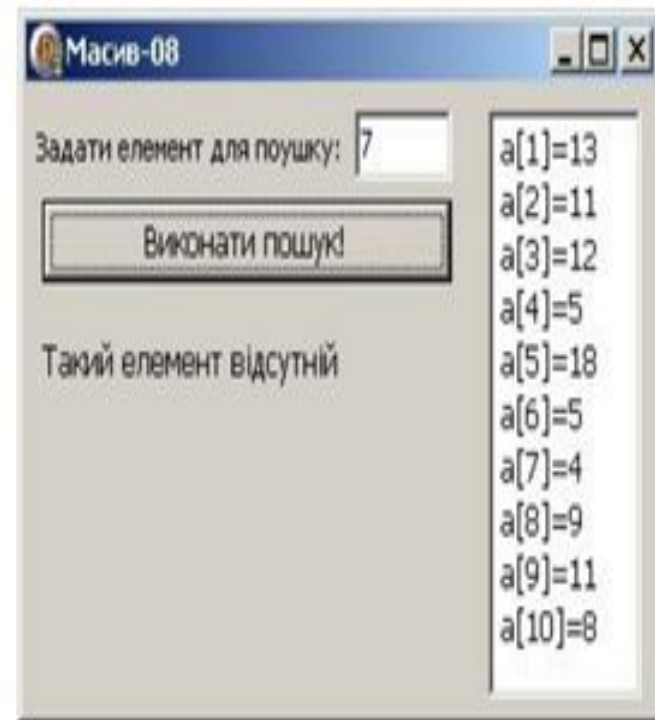
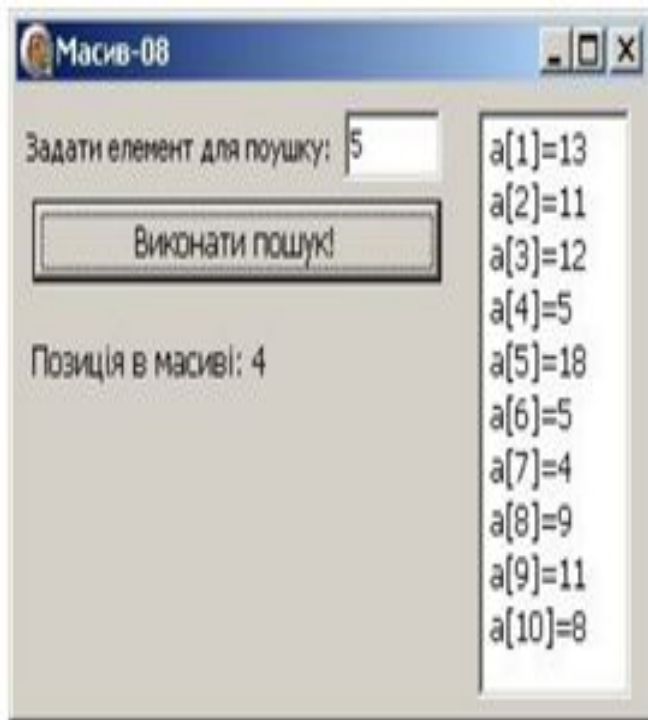
end.
```

**Пошук** – одна із дій, яка найчастіше зустрічається у задачах по програмуванню.

Вхідними даними для реалізованого алгоритму є послідовність елементів  $a[1]$ ,  $a[2]$ ,  $a[3]$ , ...,  $a[N]$  (у нашому випадку це масив) та деякий елемент  $x$ .

Задача полягає у тому, щоб отримати на виході номер елементу масиву, рівного  $x$ , або з'ясувати, що такого елементу в послідовності немає.

## Задача 6. Скласти програму для реалізації послідовного пошуку елемента у масиві (Massiv08).



```
procedure TForm1.Button1Click(Sender: TObject);
begin
    x:=strtoint (labelededit1.Text); //ввести елемент для пошук
    p:=-1; //початкове значення позиції
    for i:=1 to 10 do //перевірити всі елементи масиву
        if (a[i] = x) then //якщо знайшли елемент
            begin
                p:=i; //зберігаємо позицію
                break; //завершуємо пошук
            end;
    if p<>-1 then //якщо елемент відшукано
        label1.Caption:='Позиція в масиві: '+inttostr(p)
    else //якщо елемент відсутній
        label1.Caption:='Такий елемент відсутній';
end;
```

```
procedure TForm1.FormCreate(Sender: TObject);
begin
  randomize;
  mem1.Clear;
  for i:=1 to 10 do           //заповнити масив випадковими числами
    a[i]:=random(20);
  for i:= 1 to 10 do         //вивести елементи масиву у полі Memo
    mem1.lines.add('a['+inttostr(i)+']='+inttostr(a[i]));
end;

end.
```