

```

3. Basic
INPUT X
L = 0
M = 0
WHILE X > 0
  M = M + 1
  IF X MOD 3 <> 0 THEN
    L = L + 1
  END IF
  X = X \ 3
WEND
PRINT L
PRINT M

```

Для значэння 5637 праграма выводзіць
6 і 8.

```

4. C++
int F(int x)
{
  return x*x + 16*x + 15;
}
int main()
{
  int a, b;
  cin >> a >> b;
  int M = 0;
  for (int t = a; t <= b; t++)
    if (F(t) > 0)
      M = M + 1;
  cout << M;
  return 0;
}

```

Пры $a = -3$, $b = 5$ праграма
выводзіць 6.

3* Адлюструйце любы алгарытм з практыкавання 2 у выглядзе блок-схемы.



Глава 1

АЛГАРЫТМЫ АПРАЦОЎКІ МАСІВАЎ

§ 3. Структураваны тып даных масіў

Упершыню тып даных масіў з'явіўся ў мове Фортран (створана ў перыяд з 1954 па 1957 г. у карпарацыі ІВМ). Ужо першыя версіі мовы падтрымлівалі трохмерныя масівы (у 1980 г. максімальная размернасць масіва была павялічана да 7). Масівы былі неабходны для стварэння матэматычных бібліятэк, у прыватнасці тых, што змяшчалі працэдуры рашэння сістэм лінейных ураўненняў.

Прыклад 3.1. У 10 А класе 25 навучэнцаў. Вядомы рост кожнага ў сантыметрах. Для захоўвання значэнняў росту можна выкарыстоўваць масіў A , які складаецца з 25 цэлых лікаў. Індэкс кожнага элемента — парадкавы нумар навучэнца са спіса ў класным журнале. Тады запіс $A[5]$ — рост навучэнца пад пятым нумарам.

3.1. Паняцце масіву

У сучасным свеце штосекундна адбываецца апрацоўка вялізнай колькасці даных з дапамогай камп'ютара. Калі неабходна апрацоўваць даныя аднаго тыпу (лікі, сімвалы, радкі і інш.), то для іх захоўвання можна выкарыстаць тып даных, які называецца **масіў**.

Масіў — упарадкаваная паслядоўнасць даных, якая складаецца з канечнага ліку элементаў, што маюць адзін і той жа тып, і абазначаецца адным імем.

Масіў з'яўляецца структураваным (састаўным) тыпам даных. Гэта азначае, што велічыня, апісаная як масіў,

складаецца з канечнага ліку іншых велічынь. Так, напрыклад, можна стварыць масівы з 10 цэлых або 100 рэчаісных лікаў. Тып элементаў масіву называюць **базавым тыпам**. Усе элементы масіву ўпарадкаваны па індэксах, якія вызначаюць месцазнаходжанне элемента ў масіве.

Масіву прысвойваецца імя, пры запісе якога можна спасылацца на дадзены масіў як на адзінае цэлае. Элементы, што ўтвараюць масіў, упарадкаваны так, што кожнаму з іх адпавядае нумар (індэкс), які вызначае месца элемента ў агульнай паслядоўнасці (прыклады 3.1—3.3). Індэксы ўяўляюць сабой выразы любога простага тыпу, акрамя рэчыўнага. Доступ да кожнага асобнага элемента ажыццяўляецца зваротам да імені масіву з запісам індэкса патрэбнага элемента, індэкс элемента запісваецца пасля імені ў квадратных дужках (прыклад 3.4).

Калі зварот да элементаў масіву ажыццяўляецца пры дапамозе толькі аднаго індэкса, то такі масіў называюць **аднамерным** або **лінейным**. Элементы дадзенага масіву размяшчаюцца ланцужком адзін за адным. Колькасць індэксаў, па якіх звяртаюцца да элемента ў масіве, вызначае **размернасць масіву**. Акрамя аднамерных, могуць выкарыстоўвацца двухмерныя, трохмерныя і іншыя масівы.

3.2. Апісанне масіваў

Апісанне масіву ў мове Паскаль адбываецца наступным чынам:

```
var <імя масіву>: array [<тып індэкса>] of <тып элементаў>;
```

Прыклад 3.2. У 10 Б класе 27 навучэнцаў. У класным журнале запісаны прозвішча і імя кожнага з іх. Для захоўвання спіса навучэнцаў можна выкарыстоўваць масіў S, які складаецца з 27 радкоў. Індэкс кожнага элемента — парадкавы нумар навучэнца са спіса ў класным журнале. Тады запіс S[5] — прозвішча і імя навучэнца пад нумарам 5.

Прыклад 3.3. Кожны дзень у снежны вымяралі тэмпературу паветра. Для захоўвання значэнняў тэмпературы можна выкарыстоўваць масіў T, які складаецца з 31 рэчаіснага ліку. Індэкс элемента — нумар дня ў снежні. Запіс T[15] — тэмпература паветра 15 снежня.

Прыклад 3.4. Зварот да элемента масіва: a[3], T[i], S[n-1].

Двухмерны масіў — масіў, элементамі якога з'яўляюцца аднамерныя масівы. Яго можна ўявіць як табліцу з данымі, у якой кожны радок — лінейны масіў. Зварот да элемента ажыццяўляецца па двух індэксах: a[3][5] — элемент, змешчаны ў трэцім радку і пятым слупку. Прыкладам выкарыстання двухмернага масіву з'яўляецца ліст электроннай табліцы.

Масіў p, апісаны наступным чынам:

```
var p: array [1..30] of array [1..30] of boolean;
```

можна выкарыстоўваць для вызначэння свабодных месцаў у глядзельнай зале. Тады запіс

```
p[2][13] := true;
```

будзе абазначаць, што ў другім радзе месца 13 свабоднае, а запіс

```
p[5][7] := false; —
```

у пятым радзе месца 7 занятае.

Прыклад 3.5. Апішам масіў, разгледжаны ў прыкладзе 3.1. Памер апісанага масіву — 25 элементаў:

```
var A: array[1..25] of integer;
```

Прыклад 3.6. Апішам масіў, разгледжаны ў прыкладзе 3.2:

```
var S: array[1..27] of string;
```

Прыклад 3.7. Апішам масіў, разгледжаны ў прыкладзе 3.3. Памер апісанага масіву — 31 элемент.

```
var T: array[1..31] of real;
```

Прыклад 3.8*. Апісаць масіў для захоўвання наступных даных: маецца будынак склада, у якім ёсць два падземныя паверхі, цокальны і тры верхнія. Неабходна захоўваць колькасць пустых адсекаў склада на кожным паверсе. Памер апісанага масіву — 6 элементаў:

```
const verh = 3;
      niz = -2;
var Sklad: array[niz..verh] of
                    integer;
```

Прыклад 3.9. Каманда `b := a`; дапушчальная для масіваў `a` і `b`, апісаных наступным чынам:

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

Але каманда `b := a`; выдасць памылку, калі масівы будуць апісаны так:

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

або так:

```
var a: array[1..10] of integer;
    b: array[1..15] of integer;
```

або так:

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

Імя масіву з’яўляецца ідэнтыфікатарам і задаецца па тых жа правілах, што і імёны любых іншых пераменных.

Тып індэкса вызначае, як будуць нумаравацца элементы ў масіве. Для задання тыпу індэкса паказваюць нумар першага элемента ў масіве, затым ставяць дзве кропкі, пасля якіх пішуць нумар апошняга элемента. Даныя, якія выкарыстоўваюцца для задання індэксаў, павінны быць канстантамі. Дыяпазон індэксаў вызначае максімальна магчымая колькасць элементаў у масіве — **памер масіву**.

Тып элементаў задае значэнне базавага тыпу для дадзенага масіву. Базавы тып можа быць любым з вядомых вам тыпаў (прыклады 3.5—3.8).

3.3. Аперацыі над масівамі

Масівы, апісаныя аднолькава (у адной камандзе апісання), можна выкарыстоўваць у аперацыях прысвойвання. У выніку выканання гэтай каманды ўсе элементы аднаго масіву будуць перапісаны ў другі (прыклад 3.9). Калі масівы апісаны аднолькава, але ў розных радках ці апісаны парознаму, то пры спробе прысвойвання ўзнікне памылка пра немагчымасць пераўтварыць тыпы.

Ніякія іншыя аперацыі для масіву як для тыпу даных не вызначаны.

Аперацыі, якія выконваюцца з элементамі масіву, адпавядаюць аперацыям, выкананым над базавым тыпам. Калі, напрыклад, апісаны масіў з лікаў тыпу `integer`, то з элементамі такога масіву можна выконваць такія ж аперацыі, як і з цэлымі ліка-

мі. Элементы масіву называюць **індэксаванымі пераменнымі**. Яны могуць выкарыстоўвацца так жа, як і простыя пераменныя (прыклад 3.10).

3.4. Увод і вывад элементаў масіву

Каб працаваць з масівам, неабходна задаць пачатковыя значэнні элементаў масіву. Зрабіць гэта можна некалькімі спосабамі:

1) увод элементаў масіву з клавіятуры;

2) выкарыстанне выпадковых лікаў для вызначэння значэнняў;

3) выкарыстанне функцый (стандартных ці ўласных) для вызначэння значэнняў;

4) вызначэнне элементаў масіву як канстант.

Пры ўводзе элементаў масіву з клавіятуры кожны элемент павінен уводзіцца асобна. Калі колькасць элементаў, якія ўводзяцца, вызначана, то можна выкарыстаць цыкл **for** (прыклад 3.11).

Пры ўводзе элементаў масіву неабходна памятаць, што колькасць элементаў, якія ўводзяцца, не можа быць большай за памер масіву. У масіў, апісаны ў прыкладзе 3.11, можна ўвесці любую колькасць лікаў ад 1 да 10, змяніўшы значэнне 10 у загалоўку цыкла.

Пры апісанні масіву памер вызначае максімальную колькасць магчымых элементаў. Пры ўводзе можна вызначыць колькасць элементаў, якая неабходна для апрацоўкі ў кожным пэўным выпадку (прыклад 3.12).

Прыклад 3.9. Працяг.

```

PascalABC.NET
Файл Правка Вид Программа Сервис Модули Помощь
aProgram1.pas
var a, b : array [1..10] of integer;
    c: array [1..10] of integer;
begin
    for var i := 1 to 10 do
        a[i]:=i;
    b := a;
    for var i := 1 to 10 do
        write(b[i], ' ');
    c := b;
Список ошибок
Строка Описание Файл
1 9 Нельзя преобразовать тип array [1..10] of integer к... Pr3_9.pas

```

Прыклад 3.10. Аперацыі над індэксаванымі пераменнымі:

```

a[3] := 25 mod 7;
s := (t[1] + t[30])/2;
a[k] := b[k]*2;
Sum := Sum + a[i];
if a[i] < 0 then ...

```

Прыклад 3.11. Увесці 10 элементаў масіву a.

```

var a: array[1..10] of integer;
begin
    writeln('Увядзіце 10 лікаў праз прабел');
    for var i := 1 to 10 do
        read(a[i]);
    ...
end.

```

Прыклад 3.12. Увесці зададзеную колькасць элементаў масіву a.

```

var a: array[1..100] of integer;
    n: integer;
begin
    writeln('Увядзіце колькасць лікаў у масіве');
    readln(n);
    writeln('Увядзіце', n, 'лікаў праз прабел!');
    for var i := 1 to n do
        read(a[i]);
    ...
end.

```

Прыклад 3.13. Увод масіву радкоў.

```
var a: array[1..100] of string;
    n: integer;
begin
  writeln('Увядзіце колькасць
    радкоў у масіве');
  readln(n);
  writeln('Увядзіце', n, 'радкоў,
    кожны з новага радка');
  for var i := 1 to n do
    readln(a[i]);
  ...
end.
```

Прыклад 3.14. Выпадковым чынам задаць n элементаў масіву a . Кожны элемент — лік з адрэзка $[0; 100]$.

```
var a: array[1..100] of integer;
    n: integer;
begin
  writeln('Увядзіце колькасць
    лікаў у масіве');
  readln(n);
  for var i := 1 to n do
    a[i] := random(101);
  ...
end.
```

Прыклад 3.15. Апісанне масіву, элементы якога з'яўляюцца лікавымі канстантамі.

```
const simple_numb: array[1..5]
  of integer = (2, 3, 5, 7, 11);
```

Прыклад 3.16. Апісанне масіву, элементы якога з'яўляюцца радковымі канстантамі.

```
const c_rgb: array of string =
  ('чырвоны', 'сіні', 'зялёны');
```

Прыклад 3.17. Вывад элементаў масіву ў слупок (па адным у слупку).

```
for var i := 1 to n do
  writeln(a[i]);
```

Прыклад 3.18. Вывад элементаў масіву ў радок (праз прабел).

```
for var i := 1 to n do
  write(a[i], ' ');
```

Калі неабходна ўводзіць масіў з радкоў, трэба памятаць, што кожны элемент уводзіцца ў асобным радку з выкарыстаннем каманды `readln` (прыклад 3.13). Выкарыстоўваць прабел як раздзяляльнік не атрымаецца, паколькі прабел будзе ўспрыняты як чарговы сімвал радка.

Часам бывае зручна задаваць элементы масіву выпадковым чынам. Для гэтага выкарыстоўваецца функцыя `random(k)`, якая генеруе выпадковы цэлы лік з прамежку $[0; k)$ (прыклад 3.14).

Калі элементы масіву павінны належаць адрэзку $[a; b]$, то можна выкарыстоўваць функцыю `random(a, b)` ці вызначыць значэнне элемента масіву так:

$$a[i] := \text{random}(b-a+1) + a;$$

Калі элементы масіву не будуць змяняцца пры рашэнні задачы, то масіў можа быць апісаны як канстанта (прыклады 3.15, 3.16). Пры такім апісанні можна не паказваць індэксы элементаў у масіве, тады нумарацыя будзе ажыццяўляцца ад нуля да колькасці элементаў у спісе мінус адзін.

Выводзіць элементы масіву можна ў слупок (прыклад 3.17) ці ў радок (прыклад 3.18). Калі элементы масіву выводзяцца ў радок, то паміж імі трэба выводзіць сімвал-раздзяляльнік (часцей за ўсё выкарыстоўваюць прабел), інакш усе лікі будуць раздрукаваны запар як адзін лік з вялікай колькасцю лічбаў. Выводзіць элементы масіву можна не толькі ў

прамым, але і ў адваротным парадку (прыклад 3.19).

3.5. Рашэнне задач

з выкарыстаннем уводу-вываду масіваў

Прыклад 3.20. Напісаць праграму, якая ўвядзе элементы масіву з клавіятуры і выведзе суму трэцяга і пятага элементаў.

Этапы выканання задання

I. Зыходныя даныя: масіў a і колькасць элементаў n .

II. Вынік: S — сума трэцяга і пятага элементаў.

III. Алгарытм рашэння задачы.

1. Увод зыходных даных.
2. Вылічэнне сумы.
3. Вывад выніку.

IV. Апісанне пераменных: a — `array[1..10] of integer`; n , S — `integer`.

Прыклад 3.21. Напісаць праграму, якая сфарміруе масіў з n лікаў з адрэзка $[0; 100]$ выпадковым чынам. Вывесці масіў на экран.

Этапы выканання задання

I. Зыходныя даныя: масіў a і колькасць элементаў n .

II. Вынік: атрыманы масіў.

III. Алгарытм рашэння задачы.

1. Увод зыходных даных.
2. Генерацыя масіву.
3. Вывад выніку.

IV. Апісанне пераменных: a — `array[1..100] of integer`; n — `integer`.

Прыклад 3.19. Вывад элементаў масіву ў радок (у адваротным парадку).

```
for var i := n downto 1 do
  write(a[i], ' ');
```

Прыклад 3.20.

V. Праграма:

```
var a: array[1..10] of integer;
    n, S: integer;
begin
  writeln('Увядзіце колькасць
         лікаў у масіве >=5');
  readln(n);
  writeln('Увядзіце ', n,
         'лікаў праз прабел!');
  for var i := 1 to n do
    read(a[i]);
  S := a[3] + a[5];
  write('Сума лікаў = ', S);
end.
```

VI. Тэсціраванне.

Окно вывода

```
Увядзіце колькасць лікаў у масіве
7
Увядзіце 7 лікаў праз прабел
12 3 4 2 1 19 7
Сума лікаў = 5
```

VII. Аналіз вынікаў. Трэці элемент масіву роўны 4, пяты элемент роўны 1, сума элементаў роўна 5.

Прыклад 3.21.

V. Праграма:

```
var a: array[1..100] of integer;
    n: integer;
begin
  writeln('Увядзіце колькасць
         лікаў у масіве');
  readln(n);
  for var i:=1 to n do
    a[i] := random(101);
  for var i := 1 to n do
    write(a[i], ' ');
end.
```

VI. Тэсціраванне:

Окно вывода

```
Увядзіце колькасць лікаў у масіве
8
66 44 80 3 100 66 3 78
```

Прыклад 3.22.

V. Праграма:

```

var a: array[1..100] of integer;
    n, k: integer;
begin
  writeln('Увядзіце колькасць
        лікаў у масіве');
  readln(n);
  for var i:=1 to n do
  begin
    a[i] := 2*random(10, 35);
    write(a[i], ' ');
  end;
  writeln;
  writeln('Увядзіце k');
  readln(k);
  write(a[k]);
end.

```

VI. Тэсціраванне:

Окно вывода

```

Увядзіце колькасць лікаў у масіве
7
50 56 66 60 32 24 66
Увядзіце k
5
32

```

Прыклад 3.23.

V. Праграма:

```

var s: array [1..20] of string;
    n, k1, k2: integer;
begin
  writeln('Колькасць навучэнцаў ');
  readln(n);
  writeln('Прозвішчы');
  for var i := 1 to n do
    readln(s[i]);
  writeln('k1 i k2');
  readln(k1, k2);
  for var i := k1 to k2 do
    writeln(s[i]);
end.

```

Прыклад 3.22. Напісаць праграму, якая сфарміруе масіў з n цотных лікаў з адрэзка $[20; 70]$ выпадковым чынам. Вывесці на экран k -ы элемент масіву.

Этапы выканання задання

I. Зыходныя даныя: масіў a і колькасць элементаў n .

II. Вынік: шуканы элемент.

III. Алгарытм рашэння задачы.

1. Увод зыходных даных.

2. Генерацыя масіву.

1.1. Каб элементы масіву былі толькі цотнымі, неабходна кожны атрыманы элемент памнажаць на 2.

1.2. Паколькі элементы памнажаюцца на два, межы зыходнага адрэзка трэба паманшыць у два разы.

1.3. Вывад масіву па элементах.

3. Увод значэння k і вывад выніку.

IV. Апісанне пераменных: a — `array[1..100] of integer`; n , k — `integer`.

Прыклад 3.23. Напісаць праграму, якая ўвядзе з клавіятуры спіс прозвішчаў навучэнцаў і выведзе з яго прозвішчы з нумарамі ад k_1 да k_2 .

Этапы выканання задання

I. Зыходныя даныя: масіў s і колькасць навучэнцаў n , нумары прозвішчаў — k_1 і k_2 .

II. Вынік: спіс зададзеных прозвішчаў.

III. Алгарытм рашэння задачы.

1. Увод зыходных даных.

2. Вывад выніку.

IV. Апісанне пераменных: `s — array[1..20] of string; n, k1, k2 — integer.`

Прыклад 3.24. Задаць выпадковым чынам два масівы X і Y , якія змяшчаюць па n лікаў з адрэзка $[100; 300]$, і масіў R , які змяшчае n лікаў з адрэзка $[5; 100]$. Пабудаваць на экране акружнасці, каардынаты цэнтраў якіх захоўваюцца ў масівах X і Y , а радыусы ў масіве R .

Этапы выканання задання

I. Зыходныя даныя: масівы X , Y , R і колькасць элементаў n .

II. Вынік: рысунак n акружнасцей.

III. Алгарытм рашэння задачы.

1. Увод зыходных даных.

2. Генерацыя масіваў.

3. Устаноўка празрыстага стылю заліўкі для таго, каб адлюстроўваліся акружнасці, а не кругі.

4. Вывад выніку.

IV. Апісанне пераменных: `X, Y, R — array[1..100] of integer; n — integer.`

Усе разгледжаныя вышэй спосабы ўводу і вываду масіваў універсальныя і могуць выкарыстоўвацца для розных кампілятараў мовы Pascal. У асяроддзі PascalABC.Net дадаткова рэалізаваны каманды `print` і `println`, з дапамогай якіх масіў можна вывесці без выкарыстання каманды цыкла. Каманда `print(b);` выведзе элементы масіву b у квадратных дужках праз коску: $[1,2,3,4,5,6]$.

Каманда `println` пасля вываду масіву дадаткова пераводзіць курсор на новы радок. Элементы выводзяцца гэтак жа, як і пры выкарыстанні каманды `print`.

Прыклад 3.23. Працяг.

VI. Тэсціраванне:

Окно вывода

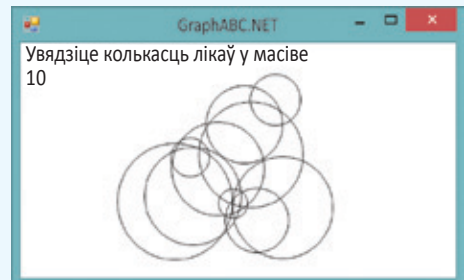
```
Колькасць навучэнцаў
6
Прозвішчы
Бялоў
Гваноў
Каралёў
Пятроў
Сідараў
Яшкін
k1 і k2
3
5
Каралёў
Пятроў
Сідараў
```

Прыклад 3.24.

V. Праграма:

```
uses graphABC;
var X, Y, R: array[1..100]
    of integer;
    n: integer;
begin
    SetWindowSize(400,400);
    writeln('Увядзіце колькасць
        лікаў у масіве');
    readln(n);
    writeln(n);
    for var i := 1 to n do
    begin
        X[i]:= random(100,300);
        Y[i]:= random(100,300);
        R[i]:= random(5,100);
    end;
    SetBrushStyle(bsClear);
    for var i := 1 to n do
        circle(X[i],Y[i],R[i])
    end.
```

VI. Тэсціраванне.





1. Што такое масіў?
2. Як апісваюцца масівы?
3. Што такое памер масіву?
4. Якія аперацыі дапушчальныя для масіваў?
5. Якія спосабы задання значэнняў элементам масіву вы ведаеце?
6. Як можна вывесці масіў?



Практыкаванні

- 1 Выкарыстоўваючы прыклады 3.14—3.18, выканайце наступныя заданні.
 1. Увядзіце 5 лікаў і выведзіце іх у адным радку.
 2. Увядзіце 7 лікаў і выведзіце іх у адным радку ў адваротным парадку.
 3. Задайце 10 выпадковых лікаў і выведзіце іх па адным у радку.
 4. Выведзіце на экран элементы масіву, зададзенага ў прыкладзе 3.16.
- 2 Змяніце праграму з прыкладу 3.19 так, каб выводзіўся здабытак першых трох элементаў.
- 3 Выкарыстоўваючы праграмы з прыкладу 3.19 або 3.20, задайце масіў з n выпадковых лікаў з адрэзка $[-10; 10]$. Выведзіце: першы элемент; апошні элемент; элемент, які стаіць на сярэднім месцы.
- 4 Увядзіце масіў з n радкоў з клавіятуры. Выведзіце элементы масіву ў адваротным парадку.
- 5 Для масіву, апісанага ў прыкладзе 3.2, увядзіце даныя з клавіятуры. Задайце нумар навучэнца. Выведзіце яго прозвішча.
- 6* Увядзіце рост навучэнцаў свайго класа, арганізаваўшы ўвод наступным чынам:

Увядзіце колькасць навучэнцаў у класе: 15
 Уводзьце рост навучэнцаў
 навучэнец нумар 1: 165
 навучэнец нумар 2: 170
 навучэнец нумар 3: 156
- 7* Для масіву, апісанага ў прыкладзе 3.3, задайце значэнні выпадковымі рэчыўнымі лікамі з інтэрвалу $(-20; 10)$. Выведзіце значэнні тэмператур для зададзенага дыяпазону дат. Прыклад вываду для дыяпазону дат ад 1 снежня да 8 снежня:
 - 1 снежня тэмпература была = 9.4
 - 2 снежня тэмпература была = -11.8
 - 3 снежня тэмпература была = -16.6
 - 4 снежня тэмпература была = 8
 - 5 снежня тэмпература была = 0.9
 - 6 снежня тэмпература была = -9.3
 - 7 снежня тэмпература была = -11.5
 - 8 снежня тэмпература была = 6.6
- 8 Змяніце праграму з прыкладу 3.24 так, каб акружнасці маляваліся рознымі колерамі.