

§ 4. Выкананне арыфметычных дзеянняў над элементамі масіву

4.1. Вылічэнне сум і здабыткаў элементаў масіву

Аперацыі, якія выконваюцца з элементамі масіву, адпавядаюць аперацыям, якія выконваюцца над базавым тыпам элементаў масіву (прыклад 4.1).

Прыклад 4.2. Зададзены аднамерны масіў з цэлых лікаў. Знайсці суму і здабытак элементаў гэтага масіву.

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

II. Вынік: S — сума элементаў і P — здабытак элементаў масіву.

III. Алгарытм рашэння задачы.

1. Увод зыходных даных. Масіў уводзіцца паэлементна з клавіятуры.

2. Вызначэнне пачатковага значэння для сумы ($S := 0$) і для здабытку ($P := 1$).

3. У цыкле дабаўляем чарговы элемент масіву да сумы і да здабытку.

4. Вывад выніку.

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

Прыклад 4.3. Вядомыя адзнакі па інфарматыцы ўсіх навучэнцаў 10 Б класа за першую чвэрць. Паспяховасць у класе будзем лічыць добрай, калі сярэдні бал большы за 7, дрэннай, калі сярэдні бал ніжэйшы за 4, у астатніх выпадках — паспяховасць сярэдня. Вызначыць паспяховасць класа па зададзеных адзнаках.

I. Зыходныя даныя: масіў a для захоўвання адзнак і колькасць навучэнцаў n .

Прыклад 4.1.

Калі базавым тыпам элементаў масіву з'яўляецца тып `integer`, то для элементаў масіву дапушчальныя наступныя аперацыі: `+`, `-`, `*`, `div`, `mod`.

Калі ў масіве захоўваюцца лікі тыпу `real`, то дапушчальнымі будуць аперацыі `+`, `-`, `*`, `/`.

Калі ў масіве захоўваюцца радкі, то для кожнага яго элемента дапушчальныя радковыя функцыі і працэдуры.

Прыклад 4.2.

V. Праграма:

```
var a: array[1..10] of integer;
    n, S, P: integer;
begin
  write('Увядзіце n = ');
  readln(n);
  writeln('Увядзіце элементы');
  for var i := 1 to n do
    read(a[i]);
  S := 0;
  P := 1;
  for var i := 1 to n do
    begin
      S := S + a[i];
      P := P * a[i];
    end;
  writeln('Сума = ', S);
  writeln('Здабытак = ', P);
end.
```

VI. Тэсціраванне.

```
Окно вывода
Увядзіце n = 5
Увядзіце элементы
3 2 44 -1 3
Сума = 51
Здабытак = -792
```

VII. Аналіз вынікаў. Правярыць правільнасць вылічэнняў можна на калькулятары.

Прыклад 4.3.

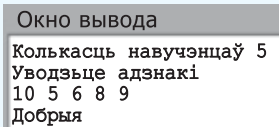
V. Праграма:

```

var a: array[1..30] of integer;
    n, S: integer; Sr: real;
begin
write('Колькасць навучэнцаў ');
readln(n);
writeln('Уводзьце адзнакі');
for var i := 1 to n do
  read(a[i]);
S := 0;
for var i := 1 to n do
  S := S + a[i];
Sr := S / n;
if Sr > 7 then
  writeln('Добрая')
else
  if Sr < 4 then
    writeln('Дрэнная')
  else
    writeln('Сярэдняя');
end.

```

VI. Тэсціраванне.



```

Окно вывода
Колькасць навучэнцаў 5
Уводзьце адзнакі
10 5 6 8 9
Добрыя

```

VII. Аналіз вынікаў.

Прыклад 4.4.

V. Праграма:

```

var Kol, Cen: array[1..50] of
  integer;
    n, Sum: integer;
begin
write('Увядзіце колькасць
  відаў тавараў ');
readln(n);
for var i := 1 to n do
begin
  writeln('Увядзіце колькасць
    тавару', i, ' і яго цану ');
  read(Kol[i], Cen[i]);
end;
Sum := 0;
for var i := 1 to n do
  Sum := Sum + Kol[i]*Cen[i];
writeln('Сумарны кошт
  тавараў =', Sum);
end.

```

II. Вынік: адно са слоў — «добрая», «сярэдняя», «дрэнная» ў залежнасці ад значэння сярэдняга бала.

III. Алгоритм рашэння задачы.

1. Увод зыходных даных. Спачатку ўводзім колькасць навучэнцаў у класе, затым масіў адзнак (паэлементна з клавіятуры).

2. Для вызначэння паспяховасці трэба вылічыць сярэдні бал (пераменная Sr). Сярэдні бал вызначаецца як сума (пераменная S) усіх адзнак, падзеленая на колькасць навучэнцаў у класе. Пачатковае значэнне для сумы — S := 0.

3. У цыкле дабаўляем чарговы элемент масіву да сумы.

4. Дзелім атрыманую суму на колькасць навучэнцаў у класе.

5. Правяраем значэнне сярэдняга бала і выводзім вынік.

IV. Апісанне пераменных: a — array[1..30] of integer; n, S — integer; Sr — real.

4.2. Вылічэнне сум і здабыткаў пры рабоце з двума масівамі

Прыклад 4.4. На складзе захоўваюцца тавары. Для кожнага віду тавару вядома колькасць адзінак тавару і кошт за адзінку тавару. Вызначыць сумарны кошт усіх тавараў, якія захоўваюцца на складзе.

I. Зыходныя даныя: Cen — аднамерны масіў для захоўвання кошту адзінкі тавару кожнага віду, Kol — масіў для захоўвання колькасці тавару кожнага віду, n — колькасць відаў тавараў.

II. Вынік: Sum — значэнне сумарнага кошту тавараў на складзе.

III. Алгарытм рашэння задачы.

1. Увод зыходных даных. Для кожнага віду тавару задаецца яго цана і колькасць.

2. Кошт усіх тавараў аднаго віду вызначаецца як здабытак колькасці на кошт. Сумарны кошт — сума ўсіх такіх здабыткаў. Пачатковае значэнне сумы $Sum := 0$. У цыкле да сумы дадаюцца здабыткі $Kol[i]*Cen[i]$.

3. Вывад выніку.

IV. Апісанне пераменных: Cen, Kol — `array[1..50] of integer`; n, Sum — `integer`.

4.3*. Выкарыстанне масіву, элементы якога з'яўляюцца канстантамі

Прыклад 4.5. Зададзены натуральны лік n ($n < 5000$). Вызначыць, ці з'яўляецца гэты лік простым.

I. Зыходныя даныя: n — натуральны лік.

II. Вынік: вывад паведамлення «просты» ці «састаўны».

III. Алгарытм рашэння задачы.

1. Увод зыходных даных.

2. Вядома, што лік n з'яўляецца простым, калі ён не дзеліцца ні на адзін просты лік, не большы за \sqrt{n} . Максімальны лік па ўмове — 5000, $\sqrt{5000} \approx 70,7107$. Створым масіў канстант s_n з простых лікаў, не большых за 71.

3. У цыкле будзем дзяліць лік n на кожны з лікаў, не большы за \sqrt{n} , якія захоўваюцца ў масіве

Прыклад 4.4. Працяг.

VI. Тэсціраванне.

Окно вывода

```
Увядзіце колькасць відаў тавараў 3
Увядзіце колькасць тавару 1 і яго цану
5 2
Увядзіце колькасць тавару 2 і яго цану
7 3
Увядзіце колькасць тавару 3 і яго цану
4 5
Сумарны кошт тавараў = 51
```

VII. Аналіз вынікаў.

Прыклад 4.5.

Умова $s_n[i] \leq \sqrt{n}$ правяраецца доўга за кошт выкліку функцыі \sqrt{n} . Гэту ўмову звычайна замяняюць эквівалентнай: $s_n[i]*s_n[i] \leq n$.

V. Праграма:

```
const s_n: array of integer =
(2, 3, 5, 7, 11, 13, 17, 19, 23, 29,
1, 37, 41, 43, 47, 53, 59, 61, 67, 71);
var n,i: integer;
begin
  writeln('Увядзіце лік');
  read(n);
  i := 0;
  while (s_n[i] * s_n[i] <= n
    and (n mod s_n[i] <> 0)) do
    i := i + 1;
  if s_n[i]*s_n[i] > n then
    writeln('Просты')
  else
    writeln('Састаўны')
end.
```

VI. Тэсціраванне.

Окно вывода

```
Увядзіце лік
2027
Просты
```

Окно вывода

```
Увядзіце лік
2021
Састаўны
```

VII. Аналіз вынікаў. Правярыць правільнасць вылічэнняў можна на калькулятары або паглядзець у табліцы простых лікаў¹.

¹ Тэхнічныя табліцы:

<http://tehtab.ru/guide/guidemathematics/guidemathematicsfigurestable/simplefigures/>
(дата доступу: 10.02.2019).

Прыклад 4.6.

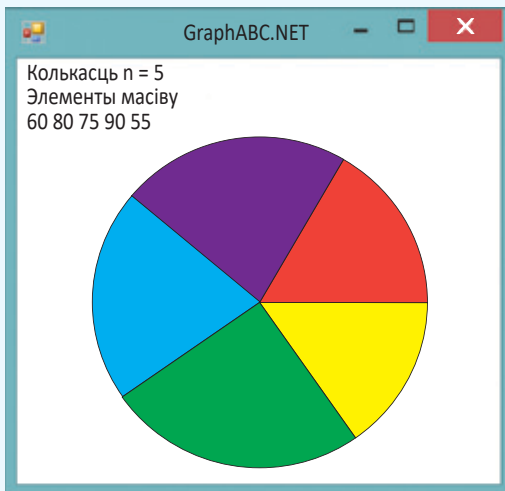
V. Праграма:

```

uses graphABC;
var a: array[1..10] of integer;
    n, S, u0, u1: integer;
begin
write('Колькасць n = ');
readln(n);
writeln(n);
writeln('Элементы масіва');
for var i := 1 to n do
read(a[i]);
for var i := 1 to n do
write(a[i], ' ');
S := 0;
for var i := 1 to n do
S := S + a[i];
u0 := 0;
for var i := 1 to n do
begin
u1:= u0+trunc(a[i]*360/S);
SetBrushColor(clRandom);
Pie(150,150,100,u0,u1);
u0 := u1;
end;
end.

```

VI. Тэсціраванне.



канстант. Калі лік n не падзяліўся ні на адзін з іх, то ён — просты, інакш — састаўны.

4. Правяраем, з якой умовай скончыў работу цыкл: лік з'яўляецца простым, калі апошні прагледжаны элемент масіва большы за \sqrt{n} (лік ні на што не падзяліўся).

5. Вывад выніку.

IV. Апісанне пераменных: s_n — const array of integer; n, i — integer.

4.4. Пабудова кругавой дыяграмы

Прыклад 4.6. Зададзены аднамерны масіў з цэлых лікаў. Пабудаваць кругавую дыяграму па лікавых даных, якія захоўваюцца ў масіве. Напрыклад, для 5 элементаў масіва — 60, 80, 75, 90, 55.

I. Зыходныя даныя: масіў a для захоўвання даных і n — колькасць даных.

II. Вынік: кругавая дыяграма.

III. Алгоритм рашэння задачы.

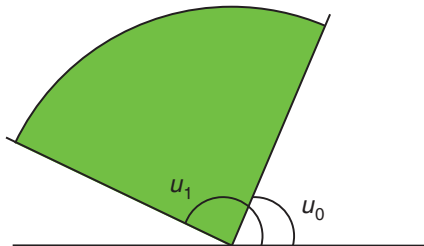
1. Увод зыходных даных. Масіў уводзіцца паэлементна з клавіятуры.

2. Кругавая дыяграма складаецца з n сектараў. Градусная мера сектара вызначаецца лікавым значэннем адпаведнага элемента ў масіве. Сумарнае значэнне ўсіх элементаў масіва (пераменная S) адпавядае велічыні ў 360° . Тады значэнню элемента масіва $A[i]$ будзе адпавядаць велічыня —

$$A[i] * \frac{360}{S}.$$

3. Вылічваем суму ўсіх элементаў масіву.

4. У цыкле будзем сектары, градусная мера якіх роўная цэлай частцы велічыні $A[i] * \frac{360}{S}$.



Для пабудовы сектара трэба ведаць велічыні двух вуглоў: u_0 і u_1 . Значэнне $u_1 = u_0 + A[i] * \frac{360}{S}$. Спачатку $u_0 = 0$. Затым у цыкле мяняем значэнне u_0 на u_1 . Колер сектара будзем задаваць выпадковым чынам. Для вылічэння цэлай часткі можна выкарыстоўваць функцыі `trunc` і `round`.

IV. Апісанне пераменных: `a` — `array[1..10] of integer`; `n`, `S`, `u_0`, `u_1` — `integer`.

- 1. Якія аперацыі дапушчальныя для элементаў масіву цэлых лікаў?
- 2. Якія аперацыі дапушчальныя для элементаў масіву рэчывых лікаў?
- 3. Як запісаць даныя ў масіў канстант?

Практыкаванні

- 1 Для задачы з прыкладу 4.2 выканайце пералічаныя заданні.
 - 1. Запоўніце табліцу.

	n	a	S	P
1	3	-2 -3 -5		
2	5	1 2 3 4 5		
3	10	1 -3 -2 3 4 3 2 4 3 2		

Прыклад 4.6. Працяг.

VII. Пабудуйце па гэтых даных дыяграму ў Excel і параўнайце.

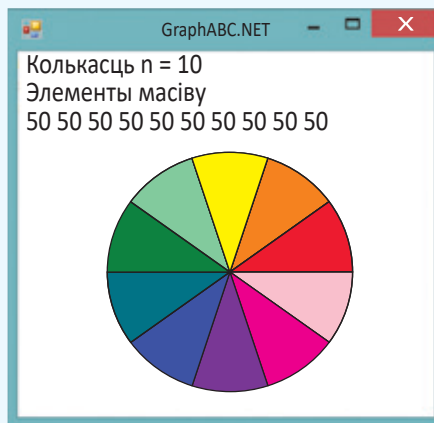
Для задання колеру сектара можна выкарыстоўваць масіў, які змяшчае колеравыя канстанты:

```
const d_color: array [1..10]
of Color = (clRed, clOrange,
clYellow, clLightGreen, clGreen,
clTeal, clBlue, clDarkViolet,
clMagenta, clPink);
```

Каманду задання колеру сектара трэба будзе замяніць на:

```
SetBrushColor(d_color[i]);
```

Вынік:



2. Дабаўце ў табліцу свае значэнні n і a .

3. Паспрабуйце падабраць такія значэнні элементаў масіву, каб $S = P$, для $n = 2, 5$.

4. Для $n = 10$ узялі ўсе элементы масіву, роўныя 9. Які вынік атрымалі? Чаму? Што трэба выправіць у праграме для атрымання правільнага выніку?

2 Для задачы з прыкладу 4.3 дабаўце вывад сярэдняга бала.

3 У ходзе хакейнага матча выдаляліся гульцы абедзвюх каманд. Для кожнага выдаленага гульца вядомы час яго адсутнасці на полі. Вызначыце, якая з каманд правяла больш часу на лаўцы штрафнікоў.

4 Для задачы з прыкладу 4.5 выканайце наступнае заданне:

Увядзіце лік 5557. Чаму з'явілася памылка? Дапоўніце масіў канстант простымі лікамі так, каб праграма магла выдаваць адказ для лікаў, меншых за 10 000. (Для гэтага можна выкарыстаць саму праграму або табліцу простых лікаў.)

5 Для задачы з прыкладу 4.6 выканайце пералічаныя заданні.

1. Унясіце ў праграму змяненні так, каб колер сектара выбіраўся з масіву канстант.

2*. Змяніце праграму так, каб дыяграма заўсёды будавалася ў цэнтры графічнага акна. Дыяметр круга вызначаецца меншай з дзвюх велічынь — шырынёй або вышынёй акна.

6* У масівах X і Y захоўваюцца каардынаты пунктаў. Пабудуйце многавугольнік, заданы гэтымі каардынатамі. Запытайце ў карыстальніка нумары двух пунктаў і пабудуйце дыяганаль многавугольніка, якая злучае гэтыя пункты.

§ 5. Пошук элементаў з зададзенымі ўласцівасцямі

Чалавек увесь час сутыкаецца з задачамі пошуку патрэбнай інфармацыі. Тыповым прыкладам можа служыць праца з даведнікамі ці бібліятэчнай картатэкай. У сучасным свеце інфармацыю шукаюць з выкарыстаннем сеткі Інтэрнэт.

Каб пошук быў выніковым і хуткім, распрацоўваюць эфектыўныя алгарытмы пошуку. Важную ролю ў працэсе пошуку інфармацыі адыгрывае спосаб захоўвання даных. Адной з самых простых структур для гэтага з'яўляецца масіў.

5.1. Лінейны пошук

Разгледзім, як ажыццяўляецца пошук для даных, што захоўваюцца ў масіве.

Сярод разнавіднасцей найпрасцейшых задач пошуку, якія сустракаюцца на практыцы, можна вылучыць наступныя тыпы:

1. Знайсці хоць бы адзін элемент, роўны зададзенаму элементу X . У выніку неабходна атрымаць i — індэкс