

Тема №1 4.

ГРАФІЧНІ МОЖЛИВОСТІ МОВИ PASCAL

Для виводу графічних зображень на екран Pascal надає користувачу бібліотеку *Graph*. Загальний вид програми з використанням *Graph* має вигляд:

```
PROGRAM <ім'я>;
USES Graph;
VAR
    grDriver, grMode, errCode: integer;
BEGIN
grDriver:= Detect; {Визначення номеру встано-
    вленого в системі драйверу}
initGraph(grDriver, grMode, ""); {ініціалізація
    графічного режиму}
if errCode= grOK then
    begin {режим відкритий і можна працювати}
    end
else
    begin {графічний режим не вдалося відкри-
        ти, проаналізуйте вміст змінної errCode}
    end;
END.
```

Двоє лапок у *initGraph* закономірні лише тоді, коли графічна програма знаходиться в тому самому каталозі, що і бібліотека *Graph*. В протилежному випадку в лапках повинен бути зазначений шлях до бібліотеки *Graph* (до файлу *EgaVga.bgi*).

Модуль *Graph* містить велику кількість описів процедур і функцій, які надають можливість працювати з кольорами, текстом, відображати фігури, лінії, будувати точки і т.ін. Всі побудови відбуваються відносно екранної системи координат, яка дещо відрізняється від звичайної. Початком координат вважається верхній лівий кут екрана, вісь *X* проходить зліва направо, вісь *Y* – зверху вниз.

Завдання

Скласти блок-схему алгоритму та написати програму виводу на екран графіку функції $\sin(x)$, заданої в прямокутній системі координат. Початок координат розмістити в центрі екрана, самі вісі супроводити позначками 1, -1 (вісь Y), $\pi/2$, $-\pi/2$, π , $-\pi$, $3\pi/2$, $-3\pi/2$, 2π , -2π (вісь X). Додати до створеної програми можливість вибору функції, що виводиться з наступного переліку: $\sin(x)$, $\cos(x)$, $\operatorname{tg}(x)$.

Методичні рекомендації

1. Проаналізувати технічну постановку задачі.

Виведення графіка функції здійснюється після ініціалізації графічного режиму. Побудова графіка відбувається в точках на проміжку для чого потрібно попередньо побудувати осі координат. Для того, щоб графік функції найбільш повно відображувався на екрані, потрібно так розмістити вісі ox та oy , щоб початок координат був посередині.

Екран має розміри 640×480 , тобто кінці координатних осей на екрані будуть мати значення в екранних точках, які зазначено на наступному малюнку:

(0,0)	(319,0)	(639,0)
(0,239)	(319,239)	(639,239)
(0,479)	(319,479)	(639,479)

Як видно з малюнка, вісь X не має від'ємної частини. Для того, щоб використовувати від'ємні значення X , будемо розглядати проміжок $[-319; 319]$, а при виведенні на екран до значення X бу-

демо додавати 319 (щоб перемістити точку 0 в середину екрана).

Вісь Y також не має від'ємної частини, тому до координати Y також будемо додавати значення середини екрана (239). Крім того, збільшення значень по осі Y відбувається в оберненому порядку – зверху вниз, а для нашої системи координат потрібно навпаки, тому для визначення Y будемо брати знак “-”.

Отже при виводі точки на екран, будемо зміщати початок відліку:

```
PutPixel(x+319, y+239, LightRed);
```

А значення X та Y будемо коригувати наступним чином: X зменшити на 319, Y розраховувати з мінусом. Ці значення будемо розраховувати за циклом від 0 до 639 (ширина екрана):

```
x:=i-319;
y:=-round(sin(x));
```

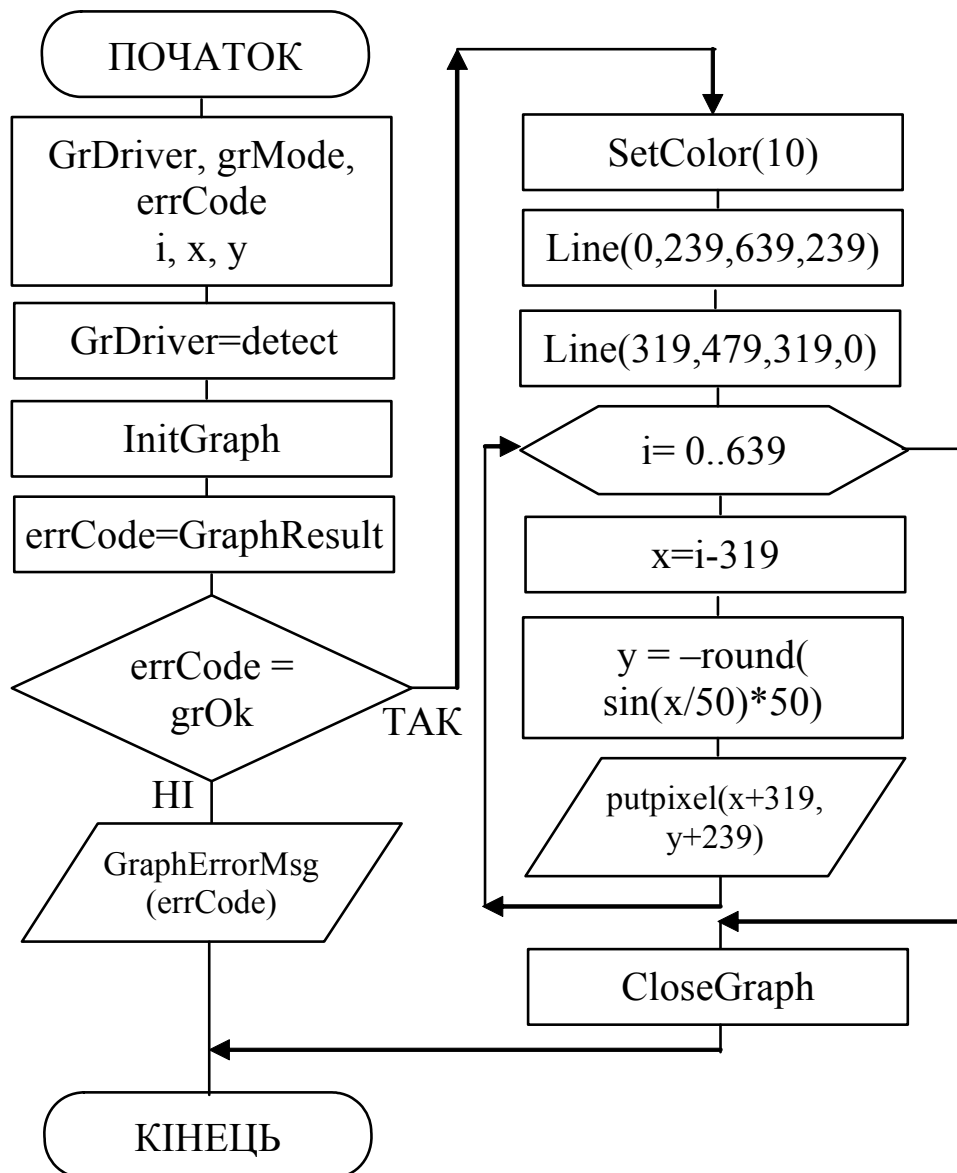
Значення $\sin(x)$ округляється, тому що екранні точки не можуть бути дробовими. Таким чином, отримаємо дуже “дрібний” графік функції, оскільки максимальне значення, яке може прийняти функція $\sin(x)-1$, один піксель, отже, щоб графік мав реальне відображення, потрібно збільшити значення функції (по Y) в декілька разів, наприклад в 50. Але тоді отримаємо невідповідність координат, для виправлення якої потрібно значення по X брати щільніше, а не лише цілі значення. Для симетрії візьмемо також в 50 разів більше, тобто Y будемо розраховувати по такій формулі.

```
y:=-round(sin(x/50)*50);
```

2. Скласти блок-схему задачі.

Програма спочатку повинна про ініціювати графічний режим, перевірити коректність, якщо при цьому виникла помилка – видати відповідне по-

відомлення і вийти з програми. Якщо графічний режим встановлено, необхідно намалювати вісі координат і вивести по точках графік функції по всій ширині екрана.



3. Записати код програми.

Відповідно до блок-схеми, програма буде такого вигляду:

```

{*****}
* GraphSin - програма побудови графіка функції *
*          sin(x)                               *
* Copyright (c) Шишук В.В. гр.ІС-04-1, 16/12/04 *
{*****}

```

```

PROGRAM GraphSin;
USES Graph;
VAR
  grDriver, grMode, errCode: integer;
  i,x,y: integer;
BEGIN
  grDriver:=detect; {автовизначення драйверу}
  InitGraph(grDriver,grMode,'h:\bp\bgi');
  {ініціалізація граф.режиму}
  errCode:=GraphResult; {результат ініціалізації}
  if errCode = grOk then {якщо успішно}
  begin
    SetColor(LightGreen); {колір осей}
    line(0,239,639,239); {вісь X}
    line(319,479,319,0); {вісь Y}
    for i:=0 to 639 do
      begin
        x:=i-319; {визначення X}
        y:=-round(sin(x/50)*50); {виз-ня Y}
        PutPixel(x+319,y+239,LightRed);
        {вивести точку на екран}
      end;
    readln; {зупинка для перегляду графіка}
    CloseGraph {закриття графічного режиму}
  end
  else
  begin
    writeln('Помилка ініціалізації графіки: ',
      GraphErrorMsg(errCode));
    writeln('Виконання програми зупинено!!!');
    readln
  end
END.

```

4. Виконати компіляцію та запустити програму на виконання.
5. Зберегти написану програму в персональну папку.
6. Змінити створену програму таким чином, щоб для осей виводилися стрілки та їх назви, а також координати початку координат, на вісі Y : точки 1 та -1 , на вісі X : $\pi/2$, $-\pi/2$, π , $-\pi$, $3\pi/2$, $-3\pi/2$, 2π , -2π .

7. Змінити програму, додаючи підпрограму, в якій відбувалася ініціалізація графічного режиму та вихід з програми при неможливості останнього.
8. Змінити програму таким чином, щоб можна було вибрати функцію з наведеного переліку ($\sin(x)$, $\cos(x)$, $\operatorname{tg}(x)$), графік якої будується.
Для організації такої можливості потрібно вивести на екран перелік необхідних функцій з номерами, зчитати з клавіатури значення, та за допомогою оператора *CASE* перевірити вибір (здійснити самостійно).
9. Записати змінену програму в персональну папку.
10. Оформити звіт про виконану роботу.

Індивідуальні завдання:

рівень **A**

Побудувати графік функції, заданої в полярній системі координат.

Варіант 1.

$$R = \frac{\pi}{\varphi}$$

Варіант 2.

$$R = 5 \sin 2\varphi$$

Варіант 3.

$$R = \sqrt{100 - \varphi^2}$$

Варіант 4.

$$R = \frac{\varphi}{\varphi + 1}$$

Варіант 5.

$$R = e^{2\varphi}$$

Варіант 6.

$$R = \operatorname{tg} \varphi \sin \varphi$$

Варіант 7.

$$R = 10 \cos 5\varphi$$

Варіант 8.

$$R = \frac{2}{\cos \varphi} + 2 \operatorname{tg} \varphi$$

Варіант 9.

$$R = 3|\sin 2\varphi|$$

Варіант 11.

$$R = \frac{\sqrt{3}}{\sin^2 \varphi}$$

Варіант 13.

$$R = \frac{10}{1 + \varphi^2}$$

Варіант 15.

$$R = \varphi \sin \varphi \cos \varphi$$

Варіант 10.

$$R = \frac{1}{\sin \varphi} - \operatorname{ctg} \varphi$$

Варіант 12.

$$R = 2\sqrt{\cos \varphi}$$

Варіант 14.

$$R = \varphi \cos^2 2\varphi$$

рівень **Б**

Зобразити на екрані процес, який описаний у відповідному варіанті.

Варіант 1.

Текст, який рухається на екрані зліва направо.

Варіант 2.

Літак, що летить, викидає бомби.

Варіант 3.

Сонце, яке “грається” промінням.

Варіант 4.

Постріли гармати.

Варіант 5.

П'ять автомобілів починають рух (по прямій) кожен по своїй лінії з різними змінними швидкостями (вид зверху).

Варіант 6.

Заливка екрана іншим кольором у вигляді ліній, які хаотично з'являються доки екран не прийме інший колір.

Варіант 7.

Людина надуває кулю насосом.

Варіант 8.

Заливка екрана у вигляді вертикальних жалюзів.

Варіант 9.

На березі річки хатинка. З димаря йде дим.

Варіант 10.

Заливка екрана у вигляді квадрату, що сходиться до центру екрана.

Варіант 11.

Надпис, що з'являється з ефектом “матриці”.

Варіант 12.

Заливка екрана у вигляді горизонтальних жалюзів.

Варіант 13.

Відображення роботи піщаного годинника.

Варіант 14.

Заливка екрана у вигляді квадрата, що з'являється в центрі екрана у вигляді крапки і потім розходить у сторони доки не заповнить весь екран.

Варіант 15.

Заливка екрана у вигляді діагонального заповнювача – з лівого нижнього кута екрана з'являється лінія, яка поступово зміщується в правий верхній кут.

рівень **B**

Зобразити на екрані процес, який описаний у відповідному варіанті.

Варіант 1.

Заставка Windows “Геометричний вальс”. Два трикутника рухаються на екрані та змінюють свій колір.

Варіант 2.

Зоряне небо, на якому спалахують та гаснуть зорі та яким пролітають комети різних величин.

Варіант 3.

В акваріумі рибка плаває в різних напрямках.

Варіант 4.

Морським дном рухається водолаз і випускає бульбашки.

Варіант 5.

Листя різних розмірів, що падає.

Варіант 6.

Циферблат годинника по якому рухаються стрілки.

Варіант 7.

Потяг, що рухається з лівого нижнього кута в верхній правий кут екрана.

Варіант 8.

Дорогою один автомобіль наздоганяє інший і обганяє його (вид зверху).

Варіант 9.

Сонячна система. Навколо сонця рухаються планети. Планета Земля з двома супутниками.

Варіант 10.

Три круга рухаються на екрані в різних напрямках, вдаряються між собою та об стінки.

Варіант 11.

Переливання води з однієї склянки в іншу.

Варіант 12.

Спортсмен, який біжить.

Варіант 13.

Підйомний кран будує багатоповерховий панельний будинок.

Варіант 14.

Дві хмари зближаються. При зіткненні – грім з блискавкою і починає йти дощ, який підсилюється. Через деякий час дощ вщухає.

Варіант 15.

В акваріумі одна велика риба-хижак наздоганяє декількох маленьких рибок і по черзі поїдає їх.

Контрольні запитання:

- 1) Яка процедура використовується для ініціалізації графічного режиму і які параметри вона має?
- 2) Наведіть всі можливі значення, які можуть приймати константи кольору для тексту та для фону?
- 3) Якою процедурою виконується установка типу ліній та тип заповнення?
- 4) Які існують можливості виведення тексту в графічному режимі?
- 5) Яким чином можна записати обробку помилок графічного режиму?
- 6) Які особливості виводу графіка функції засобами мови Pascal?
- 7) В якому файлі розміщується графічний модуль і де вказується шлях до нього?
- 8) Як можна здійснити виведення точки та лінії на екран? Які при цьому потрібно вказувати параметри?
- 9) Які функції виводять на екран стандартні фігури (квадрат, круг і т.д.)?
- 10) В чому полягає ініціювання графічного режиму? Як здійснюється вихід з графічного режиму?

ДОДАТКИ

Додаток 1.

ВІДПОВІДНІСТЬ ВІДОБРАЖЕНЬ ЧИСЕЛ У РІЗНИХ СИСТЕМАХ ЧИСЛЕННЯ

Таблиця Д1.1. Відповідність чисел у системах числення

Десяткове	Двійкове	Вісімкове	Шістнадцяткове
0	0	0	0
1	1	1	1
2	10	2	2
3	11	3	3
4	100	4	4
5	101	5	5
6	110	6	6
7	111	7	7
8	1000	10	8
9	1001	11	9
10	1010	12	A
11	1011	13	B
12	1100	14	C
13	1101	15	D
14	1110	16	E
15	1111	17	F
16	10000	20	10

Додаток 2.

ОСНОВНІ ФУНКЦІЇ СЕРЕДОВИЩА BORLAND PASCAL

Система програмування Borland Pascal являє собою сукупність компілятора мови програмування Pascal та інструментальної програмної оболонки. Для виклику Borland Pascal потрібно дати команду: BP.

Верхній рядок оболонки Borland Pascal містить «головне меню» режимів роботи, нижній рядок - коротку довідку про призначення основних функціональних клавіш. Центральна частина екрана належить вікну редактора для введення і редагування тексту програми. Під верхнім рядком у центрі подвійної рамки приводиться ім'я дискового файлу (новому файлу привласнюється ім'я NONAME00.PAS). Цифра у верхньому правому куті вікна редактора позначає номер вікна редагування. В Turbo Pascal можна одночасно працювати з 9 програмами, кожна з яких розташовується в окремому вікні редактора. Крім вікон редактора використовуються вікна: режиму налагодження, виводу результатів роботи програми, довідкової служби, стека, реєстрів.

Для переходу в головне меню необхідно натиснути клавішу F10, помістити курсор на потрібну функцію, натиснути ENTER. У спадаючому меню, що з'явиться, вибрати необхідну опцію і натиснути ENTER. Наведемо короткий опис пунктів меню інтегрованого середовища:

- *File* – виконання всіх основних операцій з файлами – створення нових, завантаження існуючих, зберігання щойно створених та відредагованих, виведення на пристрій друку вміст файлу;
- *Edit* – виконання всіх основних операцій редагування тексту – копіювання, вставка, знищення фрагментів тексту, повернення до початкового стану тексту;
- *Search* – виконання пошуку фрагментів тексту з можливістю заміни знайденого фрагмента іншим;

- *Run* – запуск програми, що знаходиться в робочій області, з можливістю покрокового виконання команд програми або її частини;
- *Compile* – здійснення компіляції програми, яка знаходиться в робочій області;
- *Debug* – виконання команд, які спрощують процес пошуку помилок в програмі (встановлення точок зупинки програми, виклик вікна відладки, вікна підпрограм, що використовуються програмою, вікна вихідних результатів та ін.)
- *Tools* – виконання деяких програм, не виходячи із інтегрованого середовища ТУРБО ПАСКАЛЯ;
- *Options* – встановлення необхідних для роботи параметрів компілятора та інтегрованого середовища;
- *Window* – виконання всіх основних операцій з вікнами: відкриття, закриття, переміщення, зміна розмірів;
- *Help* – отримання довідкової інформації.

Функціональні клавіші

Функціональні клавіші використовуються для швидкого керування середовищем Borland Pascal. Вони позначаються F1,..., F12 і розташовуються у верхньому рядку клавіатури.

Призначення функціональних клавіш:

- **F1** – виклик довідки;
- **F2** – зберегти програму на диск;
- **F3** – відкрити записану програму на диску у вікні редактора;
- **F4** – в режимі налагодження виконати програму до поточного положення;
- **F5** – розгорнути активне вікно на весь екран;
- **F6** – зробити активним наступне вікно;
- **F7** – покрокове виконання програми (кожен рядок окремо);

- **F8** – покрокове виконання без заходу в підпрограми;
- **F9** – компілювати програму, але не виконувати її;
- **F10** – перейти до діалогового вибору режиму роботи;
- **Ctrl-F9** – компілювати програму, завантажити її в оперативну пам'ять (або записати на диск) і виконати, після чого повернутися в середовище Borland Pascal;
- **Alt-F5** – перемінити вікно редактора на вікно виводу результатів програми;
- **Alt-X** – вихід з Borland Pascal.

Текстовий редактор середовища

Текстовий редактор призначений для створення і редагування тексту програми. Для створення тексту його вводять з клавіатури, натискаючи наприкінці кожного рядка клавішу ENTER.

Для того, щоб виправити помилки, допущені при наборі програми, необхідно перемістити курсор (невеликий миготливий прямокутник у вікні редактора) за допомогою клавіш переміщення курсору ← ↑ → ↓ у потрібне місце і виправити помилку.

Команди текстового редактора.

- **PgUp** – перемістити курсор на сторінку вгору;
- **PgDn** – перемістити курсор на сторінку вниз;
- **HOME** – перемістити курсор на початок рядка;
- **END** – перемістити курсор в кінець рядка;
- **Ctrl+PgUp** – перемістити курсор на початок тексту;
- **Ctrl+PgDn** – перемістити курсор в кінець тексту;
- **BACKSPACE** – стерти символ ліворуч від курсору;
- **DEL** – стерти символ праворуч від курсору;
- **Ctrl+Y** – стерти рядок, на якому розташовується курсор;
- **ENTER** – вставити новий рядок;
- **Ctrl+Q L** – відновити поточний рядок (діє, якщо курсор не залишав змінений рядок);

- **Ctrl+K B** – позначити початок блоку;
- **Ctrl+K K** – позначити кінець блоку;
- **Ctrl+K Y** – стерти блок;
- **Ctrl+K C** – копіювати блок;
- **Ctrl+K V** – перемістити блок;
- **Ctrl+K W** – записати блок у дисковий файл;
- **Ctrl+K R** – прочитати блок з дискового файлу;
- **Ctrl+K P** – надрукувати блок;
- **Shift+ ←,↑,→,↓,HOME,END** – швидко створити блок.

Довідкова система

В ситуації, коли потрібна довідка щодо використання тієї чи іншої функції чи оператора, або з'ясувати в чому заключається помилка в програмі, натисніть на клавішу *F1* або *CTRL-F1* (для пояснення конкретної ситуації), при цьому на екрані висвітиться необхідна довідка. У багатьох випадках довідка містить приклад невеликої програми, яку можна скопіювати у вікно редагування, запустити на виконання і отримати результат.

Додаток 3.

ЗРАЗОК ОФОРМЛЕННЯ ЗВІТУ ДО ПРАКТИЧНОГО ЗАВДАННЯ

а). Зразок титульної сторінки (для всіх завдань одна)

МІНІСТЕРСТВО ОСВІТИ ТА НАУКИ УКРАЇНИ
Європейський університет
Житомирська філія

Кафедра математики
та інформаційних технологій
група ІС-04-1

ЗВІТ

По практичних завданнях з дисципліни
“Основи програмування та алгоритмічні мови”

Виконав:



Петренко О.І.

Перевірив:

Шищук В.В.

Житомир, 2004

б) Зразок звіту для практичного завдання теми №3

Тема №3.СЕРЕДОВИЩЕ BORLAND PASCAL.
СТРУКТУРА ПРОГРАМИ. ОРГАНІЗАЦІЯ
ВВОДУ/ВИВОДУ1. Постановка задачі:

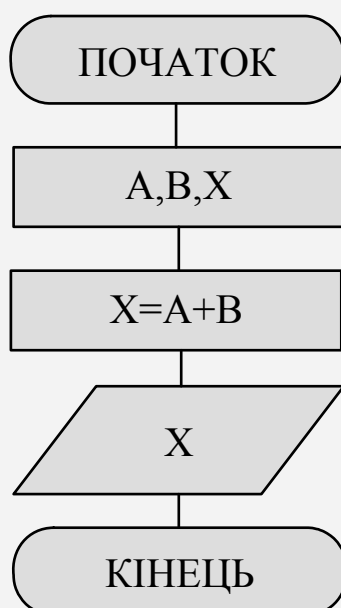
Скласти блок-схему алгоритму та створити програму задачі обчислення суми двох чисел, які задаються програмно.

2. Аналіз технічної постановки:

Для розв'язку поставленої задачі потрібно виконати просту дію додавання двох чисел, значення яких задаються у програмі.

а) *вхідні дані:* –;

б) *вихідні дані:* числа, задані програмно, та результат їх суми.

3. Блок-схема алгоритму задачі:

4. Вихідний код програми:

```
/******  
* PLUS - програма знаходження суми двох чисел, *  
* заданих програмно *  
* Copyright (c) Alex I.Petrenko IC-04-1 *  
* 17/09/2004 *  
*****/  
PROGRAM Plus;  
VAR  
    a,b,x: Integer; {оголошення змінних}  
BEGIN  
    writeln('ДОДАВАННЯ ДВОХ ЧИСЕЛ, ЗАДАНИХ В ПРО-  
            ГРАМИ');  
    writeln('-----');  
    a:=5; {присвоєння значення}  
    writeln('Перше число = ', a); {виведення a}  
    b:=7;  
    writeln('Друге число = ', b);  
    x:=a+b; {обчислення}  
    write('Результат додавання двох чисел = ');  
    writeln(x); {виведення результату}  
END.
```

5. Результат роботи програми:

```
Borland Pascal Version 7.0 Copyright (c) 1983.92 Borland International  
Перше число = 5  
Друге число = 7  
Результат додавання двох чисел = 12
```

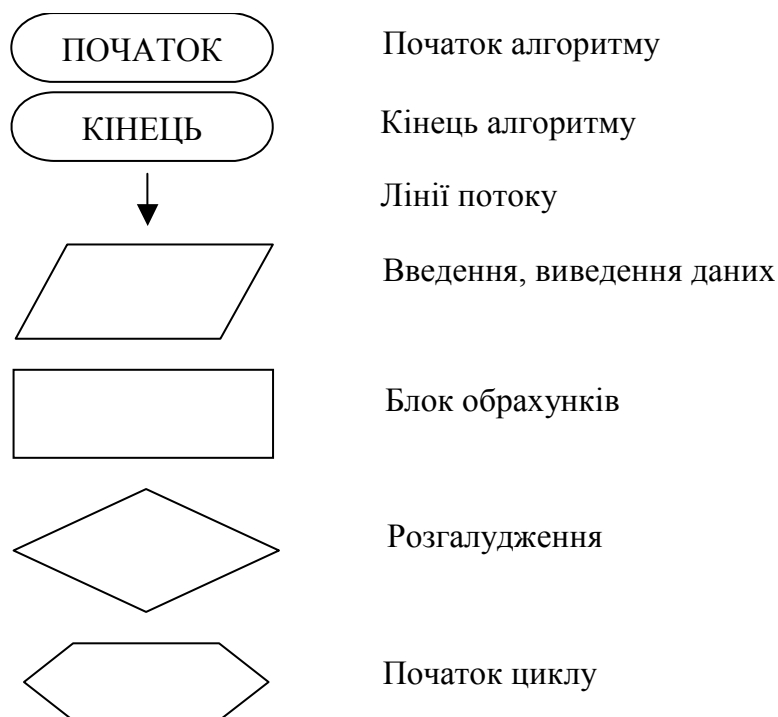
Додаток 4.

УМОВНІ ПОЗНАЧЕННЯ БЛОК-СХЕМ АЛГОРИТМІВ

Блок-схема – це спосіб представлення алгоритму в графічній формі у вигляді геометричних фігур, сполучених між собою лініями. При складанні блок-схеми алгоритму задачі використовують певний набір блоків, позначення яких мають конкретну функціональну визначеність, а текст всередині кожного блоку пояснює конкретну дію. Лінії, що сполучають блоки вказують на послідовність виконання. Лінії можуть супроводжуватися стрілками, які є обов'язковими при “незвичному” переході, тобто справа наліво або знизу доверху.

Зображення блоків, їх розміри, товщина ліній тощо, регламентується Державним стандартом “Схеми алгоритмів, програм, даних і систем” 19.701-90 (ISO 5807-85).

Нижче наведено зображення блоків з поясненням їх функціональності.





Кінець циклу



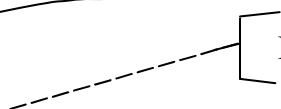
Розрив (перенесення) алгоритму



Виклик підпрограми



Виведення на принтер



Пояснення, коментар

Додаток 5.

ДИРЕКТИВИ КОМПІЛЯТОРА

У Borland Pascal можна використовувати директиви компілятора, що у вигляді оформлених коментарів особливим чином вставляються в текст програми і змінюють ті або інші можливості компілятора в процесі компіляції. Директиви обрамляються у фігурні дужки. За дужкою, що відкривається, повинний впливати знак долару \$. Знак плюс означає установку опції в активний стан, а знак мінус - у пасивний. В одній директиві можна вказати кілька опцій через кому, наприклад: {\$N+,E+}

Список основних директив компілятора:

- **{\$A-}** дозволяє для змінних і типізованих констант вирівнювання на межу слова. Для процесорів 80x86 це приводить до більш швидкого доступу до пам'яті. По замовчуванню ця опція встановлена {\$A+};
- **{\$B-}** встановлює обчислення булевих виразів за короткою схемою, що дозволяє припинити обчислення за умови отримання кінцевого результату;
- **{\$D-}** встановлює або відмінює генерацію інформації для налагодження. По замовчуванню ця опція встановлена {\$D+};
- **{\$E-}** встановлює режим емуляції співпроцесора під час роботи з дійсними даними типу SINGLE, DOUBLE, EXTENDED. По замовчуванню директива має значення {\$E+};
- **{\$I-}** виключає стандартну перевірку помилок вводу/виводу. Викликавши вбудовану функцію IOResult, програміст може сам обробляти помилки вводу/виводу. По замовчуванню ця опція встановлена {\$I+};
- **{\$N-}** встановлює режим підключення співпроцесора під час роботи з дійсними даними типу SINGLE, DOUBLE, EXTENDED. Система перевіряє наявність співпроцесора і при його відсутності команди спів-

процесора емулюються. Тому під час роботи з зазначеними дійсними даними необхідно включити дві директиви $\{ \$N+,E+ \}$;

- $\{ \$R- \}$ виключає перевірку діапазону. В момент генерації коду перевірка помилок в індексуванні масивів і в присвоєнні значень не здійснюється. По замовчуванню ця опція встановлена $\{ \$R- \}$;
- $\{ \$S- \}$ виключає перевірку стека. В момент генерації коду не здійснюється перевірка наявності достатнього простору в стеці для кожного виклику процедури або функції. По замовчуванню ця опція встановлена $\{ \$S+ \}$;
- $\{ \$V- \}$ скасовує перевірку параметрів var для рядків. Це дозволяє передавати фактичні параметри рядка, довжина яких відмінна від довжини, установленної для формального параметра var у заголовку процедури. По замовчуванню ця опція встановлена $\{ \$V+ \}$.

Додаток 6.

КОДИ ТА КОНСТАНТИ КОЛЬОРІВ

При використанні стандартних підпрограм модуля CRT для визначення кольору фону чи символів використовують коди або константи визначених кольорів.

Таблиця Д5.1 Коди та константи кольорів

Код	Константа	Колір
0	Black	Чорний
1	Blue	Темно-синій
2	Green	Темно-зелений
3	Cyan	Бірюзовий
4	Red	Червоний
5	Magenta	Фіолетовий
6	Brown	Коричневий
7	LightGray	Світло-сірий
8	DarkGray	Темно-сірий
9	LightBlue	Синій
10	LightGreen	Світло-зелений
11	LightCyan	Світло-бірюзовий
12	LightRed	Рожевий
13	LightMagenta	Малиновий
14	Yellow	Жовтий
15	White	Білий
128	Blink	Миготіння

Додаток 7.

КОДИ ТА КОНСТАНТИ ЗАПОВНЕНЬ

При використанні стандартних підпрограм модуля CRT можна визначити тип заповнення областей, для чого користуються кодами та константами заповнень.

Таблиця Дб.1 Коди та константи заповнень

Код	Константа	Заповнення
0	EmptyFill	Суцільне заповнення кольором фону (без заповнення)
1	SolidFill	Суцільне заповнення поточним кольором (що встановлений процедурою SetColor)
2	LineFill	Заповнення горизонтальними лініями
3	LtSlashFill	Заповнення похилими лініями { /// }
4	SlashFill	Заповнення похилими товстими лініями { \ \ \ }
5	BkSlashFill	Заповнення похилими товстими лініями { \ \ \ }
6	LtBkSlashFill	Заповнення похилими лініями { \ \ \ }
7	HatchFill	Заповнення перехресними лініями { +++ }
8	XhatchFill	Заповнення перехресними лініями { xxx }
9	InterleaveFill	Заповнення в клітинку
10	WideDotFill	Заповнення рідкими крапками
11	CloseDotFill	Заповнення частими крапками
12	UserFill	Шаблон заповнення, визначений користувачем

Додаток 8.

ТАБЛИЦІ СИМВОЛІВ КОДУ ASCII

Кожен символ, що може бути виведений на екран має свій код, який називається кодом ASCII. Для виводу будь – якого символу можна скористатися його кодом (за допомогою функції chr() або символу #). Нижче наведено символи (Char) їх коди в десятковій (Dec) та в шістнадцятковій системі числення (Hex).

Таблиця Д7.1. Таблиця кодів управляючих символів (0 ÷ 31)

Char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex
nul	1	00	vt	12	0B	syn	23	16
soh	2	01	ff	13	0C	etb	24	17
stx	3	02	cr	14	0D	can	25	18
etx	4	03	so	15	0E	em	26	19
eot	5	04	si	16	0F	sub	27	1A
enq	6	05	dle	17	10	esc	28	1B
ack	7	06	dcl	18	11	fs	29	1C
bel	8	07	dc2	19	12	gs	30	1D
bs	9	08	dc3	20	13	rs	31	1E
ht	10	09	dc4	21	14	us	32	1F
lf	11	0A	nak	22	15			

Таблиця Д7.2. Таблиця кодів символів з кодами 32 ÷ 127

Char	Dec	Hex	Char	Dec	Hex	Sym	Dec	Hex
пробіл	32	20	+	43	2B	6	54	36
!	33	21	,	44	2C	7	55	37
"	34	22	-	45	2D	8	56	38
#	35	23	.	46	2E	9	57	39
\$	36	24	/	47	2F	:	58	3A
%	37	25	0	48	30	;	59	3B
&	38	26	1	49	31	<	60	3C
'	39	27	2	50	32	=	61	3D
(40	28	3	51	33	>	62	3E
)	41	29	4	52	34	?	63	3F
*	42	2A	5	53	35	@	64	40

Продовження таблиці Д7.2.

Sym	Dec	Hex	Sym	Dec	Hex	Sym	Dec	Hex
A	65	41	V	86	56	K	107	6B
B	66	42	W	87	57	L	108	6C
C	67	43	X	88	58	M	109	6D
D	68	44	Y	89	59	N	110	6E
E	69	45	Z	90	5A	O	111	6F
F	70	46	[91	5B	P	112	70
G	71	47	\	92	5C	Q	113	71
H	72	48]	93	5D	R	114	72
I	73	49	^	94	5E	S	115	73
J	74	4A	_	95	5F	T	116	74
K	75	4B	`	96	60	U	117	75
L	76	4C	A	97	61	V	118	76
M	77	4D	B	98	62	W	119	77
N	78	4E	C	99	63	X	120	78
O	79	4F	D	100	64	Y	121	79
P	80	50	E	101	65	Z	122	7A
Q	81	51	F	102	66	{	123	7B
R	82	52	G	103	67		124	7C
S	83	53	H	104	68	}	125	7D
T	84	54	I	105	69	~	126	7E
U	85	55	J	106	6A	del	127	7F

Таблиця Д7.3. Кодова таблиця 866 – MS-DOS (128÷255)

Sym	Dec	Hex	Sym	Dec	Hex	Sym	Dec	Hex
А	128	80	М	140	8C	Ш	152	98
Б	129	81	Н	141	8D	Щ	153	99
В	130	82	О	142	8E	Ъ	154	9A
Г	131	83	П	143	8F	Ы	155	9B
Д	132	84	Р	144	90	Ь	156	9C
Е	133	85	С	145	91	Э	157	9D
Ж	134	86	Т	146	92	Ю	158	9E
З	135	87	У	147	93	Я	159	9F
И	136	88	Ф	148	94	а	160	A0
Й	137	89	Х	149	95	б	161	A1
К	138	8A	Ц	150	96	в	162	A2
Л	139	8B	Ч	151	97	г	163	A3

Продовження таблиці Д7.3.




Sym	Dec	Hex	Sym	Dec	Hex	Sym	Dec	Hex
д	164	A4	┆	195	C3	т	226	E2
е	165	A5	—	196	C4	у	227	E3
ж	166	A6	┆	197	C5	ф	228	E4
з	167	A7	┆	198	C6	х	229	E5
и	168	A8	┆	199	C7	ц	230	E6
й	169	A9	┆	200	C8	ч	231	E7
к	170	AA	┆	201	C9	ш	232	E8
л	171	AB	┆	202	CA	щ	233	E9
м	172	AC	┆	203	CB	ъ	234	EA
н	173	AD	┆	204	CC	ы	235	EB
о	174	AE	=	205	CD	ь	236	EC
п	175	AF	┆	206	CE	э	237	ED
	176	B0	┆	207	CF	ю	238	EE
	177	B1	┆	208	D0	я	239	EF
	178	B2	┆	209	D1	≡ / Ë	240	F0
	179	B3	┆	210	D2	± / ë	241	F1
┆	180	B4	┆	211	D3	≥	242	F2
┆	181	B5	┆	212	D4	≤	243	F3
┆	182	B6	┆	213	D5	┌	244	F4
┆	183	B7	┆	214	D6	┐	245	F5
┆	184	B8	┆	215	D7	÷	246	F6
┆	185	B9	┆	216	D8	≈	247	F7
┆	186	BA	┆	217	D9	°	248	F8
┆	187	BB	┆	218	DA	·	249	F9
┆	188	BC	■	219	DB	·	250	FA
┆	189	BD	■	220	DC	√	251	FB
┆	190	BE	■	221	DD	n	252	FC
┆	191	BF	■	222	DE	²	253	FD
┆	192	C0	■	223	DF	■	254	FE
┆	193	C1	р	224	E0		255	FF
т	194	C2	с	225	E1			

Таблица Д7.4. Кодова таблица 1251 – MS-Windows (128÷255)

Sym	Dec	Hex	Sym	Dec	Hex	Sym	Dec	Hex
Ъ	128	80	Ј	163	A3	Ж	198	C6
Ѓ	129	81	Ѡ	164	A4	З	199	C7
Ѡ	130	82	Г	165	A5	И	200	C8
ѡ	131	83	ѡ	166	A6	Й	201	C9
Ѣ	132	84	Ѕ	167	A7	К	202	CA
ѣ	133	85	Ї	168	A8	Л	203	CB
Ѥ	134	86	©	169	A9	М	204	CC
ѥ	135	87	€	170	AA	Н	205	CD
Є	136	88	»	171	AB	О	206	CE
‰	137	89	┌	172	AC	П	207	CF
Љ	138	8A	–	173	AD	Р	208	D0
<	139	8B	®	174	AE	С	209	D1
Њ	140	8C	İ	175	AF	Т	210	D2
Ћ	141	8D	°	176	B0	У	211	D3
Ѧ	142	8E	±	177	B1	Ф	212	D4
ѧ	143	8F	І	178	B2	Х	213	D5
ђ	144	90	ı	179	B3	Ц	214	D6
Ѱ	145	91	Г	180	B4	Ч	215	D7
ѱ	146	92	μ	181	B5	Ш	216	D8
“	147	93	ϱ	182	B6	Щ	217	D9
”	148	94	•	183	B7	Ъ	218	DA
•	149	95	ë	184	B8	Ы	219	DB
–	150	96		185	B9	Ь	220	DC
—	151	97	е	186	BA	Э	221	DD
□	152	98	»	187	BB	Ю	222	DE
™	153	99	Ј	188	BC	Я	223	DF
љ	154	9A	Ѕ	189	BD	а	224	E0
>	155	9B	ѕ	190	BE	б	225	E1
њ	156	9C	ı	191	BF	в	226	E2
ќ	157	9D	А	192	C0	г	227	E3
ћ	158	9E	Б	193	C1	д	228	E4
Ѡ	159	9F	В	194	C2	е	229	E5
	160	A0	Г	195	C3	ж	230	E6
ѣ	161	A1	Д	196	C4	з	231	E7
ѥ	162	A2	Е	197	C5	и	232	E8

Продовження таблиці Д7.4.

Sym	Dec	Hex	Sym	Dec	Hex	Sym	Dec	Hex
Й	233	E9	С	241	F1	Щ	249	F9
К	234	EA	Т	242	F2	Ъ	250	FA
Л	235	EB	У	243	F3	Ы	251	FB
М	236	EC	Ф	244	F4	Ь	252	FC
Н	237	ED	Х	245	F5	Э	253	FD
О	238	EE	Ц	246	F6	Ю	254	FE
П	239	EF	Ч	247	F7	Я	255	FF
Р	240	F0	Ш	248	F8			

Додаток 9.

РОЗШИРЕНІ ASCII-КОДИ

Розширені ASCII-коди повертаються тими клавішами (функціональні) чи комбінаціями клавіш, які не можна представити стандартними кодами ASCII (такі, які виводяться на екран). Розширений ASCII-код зберігається в двох байтах, перший з яких завжди містить нуль, а другий так званий scan-код. В таблицях, що наведені нижче, наведено scan-код в шістнадцятковій (Hex) та десятковій (Dec) системі числення та клавіша, натиснення якої йому відповідає (Key).

Таблиця Д8.1. Scan-коди функціональних клавіш

Key	Dec	Hex	Shift		Ctrl		Alt	
			Dec	Hex	Dec	Hex	Dec	Hex
F1	59	3b	84	54	94	5e	104	68
F2	60	3c	85	55	95	5f	105	69
F3	61	3d	86	56	96	60	106	6a
F4	62	3e	87	57	97	61	107	6b
F5	63	3f	88	58	98	62	108	6c
F6	64	40	89	59	99	63	109	6d
F7	65	41	90	5a	100	64	110	6e
F8	66	42	91	5b	101	65	111	6f
F9	67	43	92	5c	102	66	112	70
F10	68	44	93	5d	103	67	113	71

Таблиця Д8.2. Scan-коди клавіатури

Key	Dec	Hex	Key	Dec	Hex
Alt-A	30	1e	Alt-H	35	23
Alt-B	48	30	Alt-I	23	17
Alt-C	46	2e	Alt-J	36	24
Alt-D	32	20	Alt-K	37	25
Alt-E	18	12	Alt-L	38	26
Alt-F	33	21	Alt-M	50	32
Alt-G	34	22	Alt-N	49	31

Продовження таблиці Д8.2.

Key	Dec	Hex	Key	Dec	Hex
Alt-O	24	18	Alt-0	129	81
Alt-P	25	19	Alt-1	120	78
Alt-Q	16	10	Alt-2	121	79
Alt-R	19	13	Alt-3	122	7a
Alt-S	31	1f	Alt-4	123	7b
Alt-T	20	14	Alt-5	124	7c
Alt-U	22	16	Alt-6	125	7d
Alt-V	47	2f	Alt-7	126	7e
Alt-W	17	11	Alt-8	127	7f
Alt-X	45	2d	Alt-9	128	80
Alt-Y	21	15	Alt--	130	81
Alt-Z	44	2c	Alt-=	131	82
Down ↓	80	50			
Left ←	75	4b	^left	115	73
Right →	77	4d	^right	116	74
Up ↑	72	48			
NUL	3	03			
Shift-Tab	15	0f			
Del	83	53			
End	79	4f	^End	117	75
Home	71	47	^Home	119	77
Ins	82	52			
PgDn	81	51	^PgDn	118	76
PgUp	73	49	^PgUp	132	84
			^PrtSc	114	72

Таблиця Д8.3. Scan-коди розширеної клавіатури (101 клавіша)

Key	Dec	Hex	Key	Dec	Hex
F11	133	85	Alt-F11	139	8b
F12	134	86	Alt-F12	140	8c
Shift-F11	135	87	Alt-[26	1a
Shift-F12	136	88	Alt-]	27	1b
Ctrl-F11	137	89	Alt-;	39	27
Ctrl-F12	138	8a	Alt-'	40	28

Продовження таблиці Д8.3.

Key	Dec	Hex	Key	Dec	Hex
Alt-`	41	29	Alt-K /	164	a4
Alt-\	43	2b	Alt-K *	55	37
Alt-,	51	33	Alt-K -	74	4a
Alt-.	52	34	Alt-K +	78	4e
Alt-/	53	35	Alt-K Enter	166	a6
Alt-Bksp	14	0e	Ctrl-K /	149	95
Alt-Enter	28	1c	Ctrl-K *	150	96
Alt-Esc	1	01	Ctrl-K -	142	8e
Alt-Tab	165	a5	Ctrl-K +	144	90
Ctrl-Tab	148	94	Ctrl-K ↑ [8]	141	8d
Alt-up	152	98	Ctrl-K 5 [5]	143	8f
Alt-down	160	a0	Ctrl-K ↓ [2]	145	91
Alt-left	155	9b	Ctrl-K Ins [0]	146	92
Alt-right	157	9d	Ctrl-K Del [.]	147	93
Alt-Delete	163	a3			
Alt-End	159	9f			
Alt-Home	151	97			
Alt-Insert	162	a2			
Alt-PgUp	153	99			
Alt-PgDn	161	a1			

Додаток 10.

ПОВІДОМЛЕННЯ ПРО ПОМИЛКИ

Всі можливі помилки програмування в Borland Pascal 7.0 можна розділити на дві групи: синтаксичні помилки, що виявляються під час компіляції, і алгоритмічні помилки часу виконання.

Компілятор, виявивши в програмі синтаксичну помилку, встановлює курсор текстового редактора інтегрованого середовища на місце її виявлення, а також виводить повідомлення Error: з номером і найменуванням помилки.

Алгоритмічні помилки програми, що виявляються під час її виконання, приводять до того, що на екрані виникає повідомлення про помилку, а сама програма аварійно припиняє свою роботу.

Повідомлення про алгоритмічну помилку має вигляд:

```
Runtime error nnn at xxxx:yyyy.
```

Це означає: “Помилка часу виконання nnn по адресі xxxx:yyyy”, де nnn — номер помилки, а xxxx:yyyy — адреса помилки (сегмент:зсув) у завантажувальному модулі. В такій ситуації може допомогти опис кодів деяких помилок, наведених нижче.

Синтаксичні помилки

- | | |
|---|--|
| 1 | Out of memory
Вихід за межі пам'яті. |
| 2 | Identifier expected
Очікується ідентифікатор. Можливо, що ви намагаєтеся неправильно використати зарезервоване слово. |
| 3 | Unknown identifier
Ідентифікатор не описаний. |
| 4 | Duplicate identifier
Даний ідентифікатор вже використовується. |

- 5 Syntax error
Неприпустимий символ. Можливо, рядок не обмежений апострофами.
- 6 Error in real constant
Помилка в константі дійсного типу.
- 7 Error in integer constant
Помилка в константі цілого типу.
- 8 String constant exceeds line
Вихід рядка за припустимі межі. Можливо не закритий апостроф рядка.
- 10 Unexpected end of file
Несподіваний кінець файлу. Можливо, незбалансовані операторні дужки BEGIN і END чи не закритий коментар.
- 11 Line too long
Занадто довгий рядок програми.
- 12, 33 Type identifier expected
Очікується тип ідентифікатора.
- 20 Variable identifier expected
Очікується ідентифікатор змінної.
- 21 Error in type
Помилка у вказівці типу.
- 26 Type mismatch
Невідповідність типів: змінної та виразу в операторі присвоювання, фактичного і формального параметрів у звертанні до підпрограми, операндів у виразі.
- 29 Ordinal type expected
Очікується порядковий тип.
- 30..32 Constant expected
Очікується константа (ціла або дійсна).
- 34 Invalid function result type
Неприпустимий тип функції.
- 36, 37 Begin (End) expected
Очікується BEGIN (END).

- 38..40 Integer (ordinal, boolean) expression expected
Цей вираз має бути цілого (порядкового, логічного) типу.
- 41 Operand types do not match operator
Тип операндів не відповідає типу знаку операції.
- 42 Error in expression
Помилка у виразі.
- 43 Illegal assignment
Неправильне використання присвоєння.
- 50, 54, 57, 58 DO (OF, THEN, TO, DOWNT0) expected
Очікується зарезервоване слово DO (OF, THEN, TO, DOWNT0).
- 62 Division by zero
Ділення на нуль.
- 64 Cannot Read or Write variables of type
Не можна вводити чи виводити змінні цього типу.
- 74 Constant and case types do not match
Константа і тип селектора в операторі CASE несумісні.
- 76 Constant out of range
Константа виходить за межі припустимого діапазону.
- 79 Integer or real expression expected
Даний вираз повинен бути типу INTEGER або REAL.
- 85..95 ";", ":", ",", "(", ")", "=", ":", "[", or "(", ".", "]" or ".)", ".", ".." expected
Очікується зазначений розділовий знак.
- 97 Invalid FOR control variable
Неправильний параметр в операторі FOR.
- 98, 102, 103, 104, 106 Integer (String, Real, Ordinal, Character) variable expected
Ця змінна має бути цілого типу (типу STRING, типу REAL, порядкового типу, типу CHAR).

- 108 Overflow in arithmetic operation
Переповнення. Результат виходять за межі типу LONGINT.
- 112 Case constant out of range
Константа оператора CASE виходить за межі цілого типу.
- 113 Error in statement
З цього символу оператор починатися не може.
- 140 Invalid floating point operation
Дана операція над дійсними величинами призвела до переповнення чи ділення на нуль.
- 143 Invalid procedure or function reference
Неправильне звертання до підпрограми.

Алгоритмічні помилки

- 106 Invalid numeric format
Введене значення не відповідає правильному числовому формату.
- 200 Division by zero
Ділення на нуль.
- 201 Range check error
Вихід значення за межі діапазону.
- 205 Floating point overflow
Переповнення. При операції з дійсними числами отримане надто велике число.
- 207 Invalid floating point operation
Неправильна операція з дійсним числом.
- 215 Arithmetic overflow error
Вихід результату цілочислової арифметичної операції за припустимі межі.

Додаток 11.

РЕСУРСИ INTERNET, ЯКІ ПРИСВЯЧЕНІ МОВІ PASCAL

В даному додатку вказані адреси, за якими можна знайти різноманітну інформацію в мережі Internet, яка відноситься до програмування на мові Pascal. Це приклади та вихідні коди програм, навчальні посібники і статті, розсилки, та FAQ, які висвітлюють різні проблеми програмування.

<http://pascal-central.com>

Великий сайт, який містить вихідні коди програм, статті, присвячені Pascal, різну технічну інформацію, посилання на інші ресурси Internet.

<http://be-club.narod.ru/upload/library/pascal.html>

Різного роду документація, програми, модулі, які представлені у вигляді архівних файлів, доступних для скачування.

<http://pascal.sources.ru>

Вихідні коди програм, документація, FAQ, статті, різні версії дистрибутивів.

<http://citforum.ru/programming/delphi.shtml>

Статті, книжки з програмування на Pascal та Delphi.

<http://ifcity.info>

Великий сайт з різноманітною інформацією, статтями, навчальними посібниками з програмування, Інтернет-технологій, операційних систем, баз даних і т.ін. Також представлена інформація по Pascal.

<http://www.freepascal.org>

Сайт, присвячений мові Pascal, з новинами, розробками, інструментами та різними версіями компіляторів.

<http://www.programmersheaven.com/zone24/index.htm>

Великий сайт з вихідними кодами, статтями та посиланнями на ресурси цієї тематики, представлений набір інструментів розробника.

<http://web-pascal.narod.ru>

Різного роду документація, статті, версії компіляторів, розсилка, форуми та багато іншого за тематикою програмування на Pascal.

http://storage.virtualave.net/pascal_source.htm

Вихідні коди багатьох цікавих задач на Pascal та інших мовах програмування.

<http://pasc.al.ru>

Сторінка, присвячена питанням програмування на мові Pascal. Форум, тексти книжок та коди готових програм.

<http://pascal.dax.ru>

Сайт з різномірною інформацією з програмування на Pascal. Статті, лекції, довідники, задачник і т.ін., уроки, вихідні коди різної тематики, задачі та розсилка.

<http://programs-download.nm.ru/pascal.htm>

Уроки по Borland Pascal.

<http://subrins.narod.ru/pascal.html>

Методи і прийоми, які використовуються при розв'язку задач олімпіад з наведенням вихідних кодів.

<http://www.bookcase.ru/pr/prpasc1.html>

Електронна література по Pascal.

<http://rusdoc.ru/reviews/programming/pascal>

Статті з описом підходів програмування в різних версіях Pascal.

<http://programmerts.by.ru/tpascal/info>

Уроки, опис мови, вихідні коди програм.

<http://programing.narod.ru/pascal.html>

Програми на мові Pascal.

<http://gladimdim.hobi.ru/pascal/index.phtml>

Сайт зі статтями та прикладами програм, електронними посібниками з різних мов програмування.

<http://yastrebov.chat.ru/pascal.html>

Дещо для любителів Pascal. Матеріали, програми, цікаві задачі.

<http://pascal.stsland.ru/book.html>

Книжка по Borland Pascal.

<http://science.h1.ru/programs.html>

Програми на Turbo Pascal.

<http://delphiplus.nagano.ru/DocsPascal.html>

Документація по Pascal.

<http://pascal.hop.ru>

Лабораторні роботи, курсові проекти, вихідні коди програм під DOS, задачі та “маленькі хитрощі”.

<http://turbopascal.h10.ru>

Навчальний посібник по Turbo Pascal.

СПИСОК РЕКОМЕНДОВАНОЇ ЛІТЕРАТУРИ

1. Абрамов С.А. и др. Задачи по программированию. – М.: Наука, 1998.
2. Абрамов С.А., Зима Е.В., Начала программирования на языке Паскаль. – М.: Наука, 1987.
3. Алгоритмічна мова Паскаль. Основи програмування та обчислювальні методи: Навч. посібник / М.Ф.Єфремов, М.П.Решетнік, В.О.Скачков. – К.: НМК ВО, 1992.
4. Боон К. Паскаль для всех. – М.: Энергоатомиздат, 1988.
5. Бородич Ю.С., Вальвачев А.Н., Кузьмич А.И. Паскль для персональных компьютеров. Минск: Выш. шк., 1991.
6. Вирт Н. Алгоритмы + структуры данных = программы. – М.: Мир, 1985.
7. Грогоно П. Программирование на языке Паскаль. – М.: Мир, 1982.
8. Гусева А.И. Учимся программировать: Pascal 7.0. Задачи и методы их решения. – М.: Диалог МИФИ, 2001.
9. Джонс Ж., Харроу К. Решение задач в системе Турбо Паскаль. М.: Финансі и статистика, 1991.
10. Емелина Е.И. Основы программирования на языке ПАСКАЛЬ. М., 1997.
11. Епанешников А.М., Епанешников В.А. Программирование в среде TURBO PASCAL 7.0.- 3-е изд., стереотип.- М.: Диалог-МИФИ, 1997.
12. Зеленьяк О.П. Практикум программирования на Turbo Pascal. Задачи, алгоритмы и решения. – К.: ДиаСофт, 2002.

13. Информатика: Комп'ютерна техніка. Комп'ютерні технології / За ред.. О.І. Пушкаря. – К.: Видав. Центр «Академія», 2002.
14. Йенсен К., Вирт Н. Паскаль: руководство для пользователя. М., 1989.
15. Кассера Винфрид И Фолькер. Turbo Pascal 7.0 – К.: Диасофт, 2003.
16. Климов Ю.С., Касаткин А.И., Мороз С.М. Программирование в среде Turbo Pascal 6.0. – Минск: Выш.шк., 1992.
17. Климова Л.М. Практическое программирование. Решение типовых задач. – К.: КУДИЦ-ОБРАЗ, 2003.
18. Кнут Д. Искусство программирования. – Т.3: Сортировка и поиск. – М.: Вильямс, 2004.
19. Ковалюк Т.В. Основы програмування. – К.: Видавнича група ВНУ, 2005.
20. Коффман Элиот Б. Turbo Pascal, 5-е изд.: Пер. с англ. – М.: Изд. дом «Вильямс», 2002.
21. Лингер Р., Миллс Х., Уитт Б. Теория и практика структурного программирования. – М.: Мир, 1982.
22. Марченко А.И., Марченко Л.А. Программирование в среде Borland Pascal 7.0 / Под ред. В.П.Тарасенко – К.: ЮНИОР. 1998.
23. Меженный О.А. Turbo Pascal: Самоучитель. – СПб.: Диалектика, 2003.
24. Меженный О.А. Turbo Pascal: Учитесь программировать. – М.: Издательский дом «Вильямс», 2001.
25. Мизрохи С.В. TURBO PASCAL и объектно-ориентированное программирование. М., 1992.
26. Моргун А.Н. Решение задач средствами языка Turbo Pascal 7.0 – К.: Юниор, 2002.

27. Немнюгин С., Перколаб Л. Изучаем Turbo Pascal. – СПб.: Питер, 2002.
28. Немнюгин С.А. Turbo Pascal: Практикум. – СПб.: Питер, 2001.
29. Немнюгин С.А. Turbo Pascal: Программирование на языке высокого уровня: Учебник для ВУЗов. 2-е изд. – СПб.: Питер, 2005.
30. Немнюгин С.А. Turbo Pascal: Учебник. – СПб.: Питер, 2002.
31. Немнюгин С.А. Turbo Pascal: Учебник. – СПб.: Питер, 2002.
32. Окулов С.М. Программирование в алгоритмах. – М.: Бинном, Лаб. базових знаний, 2002.
33. Павловська Т.А. Паскаль. Программирование на языке высокого уровня: Учебник для вузов. – СПб.: Питер, 2005.
34. Пестриков В.М., Маслобоев А.Н. Turbo Pascal 7.0. Изучаем на примерах. – М.: Наука и Техника, 2003.
35. Пильщиков В.Н. Язык Паскаль: Упражнения и задачи. – М.: Науч. Мир, 2003.
36. Прайс Д. Программирование на языке Паскаль: Практическое руководство. М., 1987.
37. Прокудин Г.С. Основы программирования и алгоритмические языки. Основы вычислительной техники, алгоритмизации и программирования. Учебное пособие и методические указания по выполнению практических занятий. – К.: Изд-во Европ.ун-та, 2001.
38. Рапаков Г.Г., Ржеуцкая С.Ю. Программирование на языке Pascal. – К.: ВНУ, 2004.
39. Рубенкинг Н.Дж. Турбо Паскаль для Windows: в 2-х томах. Т.1. Пер. с англ. - М.: Мир, 1993.

40. Рубенкинг Н.Дж. Турбо Паскаль для Windows: в 2-х томах. Т.2. Пер. с англ. - М.: Мир, 1993.
41. Семакин И.Г., Шестаков А.П. Основы программирования: Учебник. – М.: Мастерство, 2002. – 432 с.
42. Сердюченко В.Я. Розробка алгоритмів та програмування мовою Turbo Pascal. – Харків: Паритет, 1995.
43. Ставровський А.Б. Турбо Паскаль 7.0. – К.: Издат. группа ВНУ, 2000.
44. Сухарев М. Turbo Pascal 7.0. Теорія и практика програмування. – М.: Наука и Техника, 2003.
45. Турбо Паскаль 7.0 – К.: Издательская группа ВНУ, 1999.
46. Турбо Паскаль 7.0. Самоучитель. – СПб.: Питер; К.: Издательская группа ВНУ, 2002.
47. Фараонов В.В. Турбо Паскаль 7.0. Начальный курс: Учебное пособие. – М.: «Нолидж», 1997.
48. Фаронов В.В. Turbo Pascal в подлиннике. – СПб.: ВНУ, 2003.
49. Фаронов В.В. Турбо Паскаль 7.0. Практика программирования. Учебное пособие. М., 1997.
50. Шелест В.Д. Программирование. – СПб.: БХВ Петербург, 2001.

Шищук Вадим Володимирович

**Основи програмування
на алгоритмічній мові
Pascal**

Навчальний посібник

Редактор: *Вдовиченко Валентина Миколаївна*
Коректор: *Кліменчук Євгенія Василівна*
Комп'ютерний дизайн та верстка: *Шищук Вадим Володимирович*
Дизайн обкладинки: *Сидоренко Марія Олексіївна*

Підписано до друку 16.08.2005 р. Формат 84x108/32
Друк офсетний. Папір офсетний.
Гарнітура Bookman Old Style
Умов. друк. аркушів 13,2. Обл.-вид, ар.— 10,9
Наклад 1000 прим.
Зам. №

Видавництво «Кондор»
Свідоцтво ДК № 1157 від 17.12.2002 р.
03057, м.Київ, пров. Польовий, 6,
тел./факс:(044) 456-60-82, 241-83-47