

Міністерство освіти і науки України
Житомирський державний університет імені Івана Франка
Кафедра прикладної математики та інформатики

Т.А. Вакалюк, С.С. Жуковський

**Структурне програмування
мовою Pascal
(лабораторний практикум)**

видання друге, перероблене та доповнене

*Навчальний посібник для студентів фізико-
математичного факультету*

Житомир
Вид-во ЖДУ ім. І. Франка
2011

УДК 004.42+004.432.2(076.5)

ББК 73р

В14

Затверджено вченою радою Житомирського державного університету імені Івана Франка протокол № 10 від 27.05.2011 р

Рецензенти:

Ляшенко Б.М. – доктор фізико-математичних наук, професор

Щехорський А.Й. – кандидат фізико-математичних наук, доцент

Морозов А.В. – кандидат технічних наук, доцент

В14 **Вакалюк Т.А., Жуковський С.С**

Структурне програмування мовою Pascal (лабораторний практикум). Навчальний посібник для студентів фізико-математичного факультету. – Житомир: Вид-во ЖДУ, 2011. – 120 с. (видання друге, перероблене та доповнене)

Посібник призначений для використання студентами під керівництвом викладача на лекціях, практичних та лабораторних заняттях. Посібник містить теоретичні відомості з програмування мовою Pascal, тексти лабораторних робіт (контрольні питання, завдання, вимоги до захисту лабораторних робіт), приклади виконань завдань, перелік задач для самостійного опрацювання, тести для контролю знань та умінь. Викладений матеріал відповідає діючій програмі з програмування для спеціальностей «математика та основи інформатики», «фізика та основи інформатики», «інформатика».

Для студентів фізико-математичних спеціальностей вищих педагогічних закладів, вчителів інформатики загальноосвітніх шкіл.

УДК 004.42+004.432.2(076.5)

ББК 73р

Зміст

СЕРЕДОВИЩЕ ПРОГРАМУВАННЯ FREE PASCAL

Теоретичні відомості

Лабораторна робота №1

Тестові завдання з теми «Середовище програмування Free Pascal»

ЛІНІЙНЕ ПРОГРАМУВАННЯ

Теоретичні відомості

Лабораторна робота №2

Зразок виконання лабораторної роботи

Список задач для самостійного опрацювання по темі «Лінійне програмування

Тестові завдання з теми «Лінійне програмування»

РОЗГАЛУЖЕННЯ. ОПЕРАТОР ВИБОРУ

Теоретичні відомості

Лабораторна робота №3

Зразок виконання лабораторної роботи

Список задач для самостійного опрацювання по темі «Розгалуження. Оператор вибору»

..... **Ошибка! Закладка не определена.**

Тестові завдання з теми «Розгалуження. Оператор вибору»

ЦИКЛИ. ЦИКЛ З ПАРАМЕТРОМ FOR

Теоретичні відомості

Лабораторна робота №4

Зразок виконання лабораторної роботи

ЦИКЛ З ПЕРЕДУМОВОЮ ТА ПІСЛЯУМОВОЮ

Теоретичні відомості

Лабораторна робота №5

Зразок виконання лабораторної роботи

Список задач для самостійного опрацювання по темах «Цикли. Цикл FOR, WHILE, REPEAT...UNTIL»

Тестові завдання по темам «Цикли. Цикл з параметром. Цикл з передумовою та цикл з післяумовою»

ОДНОВИМІРНИЙ МАСИВ

Теоретичні відомості

Лабораторна робота №6

Зразок виконання лабораторної роботи

ДВОВИМІРНІ МАСИВИ

Теоретичні відомості.

Лабораторна робота №7

Зразок виконання лабораторної роботи

Список задач для самостійного опрацювання по темах «Масиви. Одновимірні та двовимірні масиви»

Тестові завдання по темам «Масиви. Одновимірні та двовимірні масиви»

ПОБУДОВА ГРАФІЧНИХ ОБРАЗІВ ТА ГРАФІКІВ ФУНКЦІЙ

Теоретичні відомості.

Лабораторна робота №8

Тестові завдання по темі «Побудова графічних образів та графіків функцій»

ЛІТЕРНІ ВЕЛИЧИНИ

Теоретичні відомості

Лабораторна робота №9

Зразок виконання лабораторної роботи

Список задач для самостійного опрацювання по темі «Літерні величини»

Тестові завдання по темі «Літерні величини»

РОБОТА З ТЕКСТОВИМИ ФАЙЛАМИ

Теоретичні відомості

Лабораторна робота №10

Зразок зчитування та записування даних з файлу та у файл у двох випадках: коли відома кількість заданих чисел у файлі і коли не відома.

Тестові завдання по темі «Робота з текстовими файлами»

ВИКОРИСТАННЯ ПІДПРОГРАМ: ПРОЦЕДУР ТА ФУНКЦІЙ

Теоретичні відомості

Лабораторна робота №11

Зразки виконання задач на процедури і функції

Список задач для самостійного опрацювання по темі «Використання підпрограми: процедур та функцій»

Тестові завдання по темі «Використання підпрограм: процедур та функцій»

ІНТЕРНЕТ-ПОРТАЛ “Е-OLIMP”

Система перевірки задач в Інтернеті «e-olimp»

Зразок виконання задачі на порталі e-olimp.com

СПИСОК ЛІТЕРАТУРИ ТА ДЖЕРЕЛ

СЕРЕДОВИЩЕ ПРОГРАМУВАННЯ FREE PASCAL

Теоретичні відомості

У системі Free Pascal об'єднано **текстовий редактор** для набирання текстів програм, **компілятор** для виявлення помилок у програмах, та запуску програм на виконання в разі відсутності помилок, **налагоджувача** для покрокового виконання програми, відслідковування значень змінних для виявлення складних помилок.

Інтегроване середовище запускається файлом fp.exe. Після запуску середовища програмування ви побачите (див. малюнок) у верхній частині екрана меню, а в нижній рядок повідомлень.



Екран інтегрованого середовища програмування Free Pascal

Середовище Free Pascal дозволяє користувачу працювати як за допомогою клавіатури, так і за допомогою мишки. Для ефективності роботи в даному середовищі рекомендуємо засвоїти ряд гарячих клавіш.

F10 – активізація меню

Alt+F – активізація меню файл.

Розглянемо пункти головного меню.

File – меню для роботи файлами, яке містить наступні команди:

- **New (Alt+F -> N)**– відкрити нове вікно для створення програми.
- **Open(F3)** – відкрити з диска програму, яку було раніше збережено
- **Save(F2)** – зберегти вміст активного вікна на диску.
- **Save as..** – зберегти вміст активного вікна під новим вказаним іменем.
- **Save all** – зберегти вміст всіх активних вікон.
- **Change dir...** – зміна поточного каталога для читання або запису файлів.
- **Print** виведення на пристрій друку вміст активного вікна

- **Command shell** – тимчасовий виклик командного процесора операційної системи для виконання його команд (повернення і середовище програмування за допомогою команди EXIT).
- **Exit(Alt-X)** – Завершення сеансу роботи в інтегрованому середовищі.

Edit – редагування програми в активному вікні.

Для виділення фрагменту тексту необхідно тримаючи клавішу **Shift** переміщати курсор клавішами керування курсора. Зняти відмітку виділеного тексту за допомогою натиснення клавіш Ctrl+K+H або натисненням лівої кнопки миші.

- **Undo (Alt+BkSp)** – відмінити останню операцію.
- **Cat (Shift+Del)** – знищення виділеного фрагменту з занесенням його в буфер обміну.
- **Copy (Ctrl+Ins)** – копіювання виділеного фрагменту в буфер обміну.
- **Paste (Shift+Ins)** – вставлення фрагменту з буферу обміну в активне вікно починаючи з позиції курсора.
- **Clear (Ctrl+Del)** - знищення виділеного фрагменту без занесення його в буфер обміну.
- **Show clipboard** – поміщення вмісту буферу обміну в окреме вікно з можливістю редагування.

Run:

- **Run (Ctrl+F9)** – запуск програми на виконання з одночасним пошуком синтаксичних помилок.
- **Program reset (Ctrl+F2)** - закриття всіх відкритих програмою файлів і встановлення лічильника на початок програми.
- **Go to cursor (F4)** – запуск програми на виконання до позиції курсора.
- **Trance into (F7)** – покрокове виконання програми з входженням у всі функції.
- **Step over (F8)** - покрокове виконання програми без входженням у всі функції (виконання функції як одна операція).

Compile:

- **Compile (Alt + F9)** Компіляція активної програми.
- **Make (F9)** - перевірка синтаксичних помилок в програмі.

Debug:

- **Add watch..(Ctrl + F7)** – добавлення змінної до вікна watch для відслідковування її під час покрокового виконання програми.
- **Output** – відкриття вікна виведення.
- **User Screen (Alt+F5)** перехід на екран виведення

Window

- **Size/Move (Ctrl+F5)** – входження в режим зміни положення і розміру активного вікна. Клавіші керування курсора міняють положення вікна, Shift+ «стрілка» змінює розмір вікна.
- **Zoom (F5)** – розгорнути активне вікно на весь екран.
- **Next (F6)** – перехід до наступного вікна.
- **Previous (Shift+F6)** – перехід до попереднього вікна.
- **Close(Alt + F3)** – закрити активне вікно.
- **Close all** – закрити всі відкриті вікна.

Help:

- **Index (Shift+F1)** – відкриття алфавітний перелік ключових слів довідки або відкриття довідки по ключовому слову на якому знаходиться курсор.
- **Topic search (Ctrl+F1)** - відкриття довідки по ключовому слову на якому знаходиться курсор.
- **Previos topic (Alt+F1)** – відкриття останнього відкритого вікна довідки

Додаткові гарячі клавіши

Ctrl+Y – знищити стрічку де знаходиться курсор.

Лабораторна робота №1

Мета: Набути умінь та навички роботи в середовищі Free Pascal відкриття, виконання, введення, збереження, компіляція, налагодження програми на Free Pascal.

Матеріальне забезпечення: Free Pascal

Практичне завдання.

Варіант 0-15

1. Завантажити середовище програмування Free Pascal.
2. Набути умінь та навички роботи в середовищі Free Pascal відкриття, виконання, введення, збереження, компіляція, налагодження програми на Pascal.

Тестові завдання з теми «Середовище програмування Free Pascal»

1. За допомогою команди **New** можна:
 - 1) Створити новий файл
 - 2) Відкрити раніше створений файл
 - 3) Зберегти файл
 - 4) Закрити файл
2. За допомогою команди **Open** можна:
 - 1) Створити новий файл
 - 2) Відкрити раніше створений файл
 - 3) Зберегти файл
 - 4) Закрити файл
3. За допомогою команди **Save** можна:
 - 1) Створити новий файл
 - 2) Відкрити раніше створений файл
 - 3) Зберегти файл
 - 4) Зберегти файл під новим ім'ям
4. За допомогою комбінації клавіш **Alt+F -> N** можна:
 - 1) Створити новий файл
 - 2) Відкрити раніше створений файл
 - 3) Зберегти файл
 - 4) Закрити файл
5. За допомогою клавіші **F10** можна:
 - 1) Створити новий файл
 - 2) Активізувати меню
 - 3) Зберегти файл
 - 4) Активізувати меню файл
6. За допомогою комбінації клавіш **Alt+F** можна:
 - 1) Створити новий файл
 - 2) Активізувати меню файл
 - 3) Зберегти файл
 - 4) Закрити файл
7. За допомогою клавіші **F3** можна:
 - 1) Створити новий файл
 - 2) Активізувати меню файл
 - 3) Зберегти файл
 - 4) Відкрити файл
8. За допомогою клавіші **F2** можна:
 - 1) Створити новий файл
 - 2) Зберегти файл під новим ім'ям
 - 3) Зберегти файл

- 4) Відкрити файл
9. За допомогою команди **Save as..** можна:
 - 1) Створити новий файл
 - 2) Відкрити раніше створений файл
 - 3) Зберегти файл
 - 4) Зберегти файл під новим ім'ям
10. За допомогою команди **Exit** можна:
 - 1) Завершити сеанс роботи в інтегрованому середовищі
 - 2) Редагувати програми в активному вікні
 - 3) Зберегти файл
 - 4) Зберегти файл під новим ім'ям
11. За допомогою команди **Undo** можна:
 - 1) знищити виділений фрагмент з занесенням його в буфер обміну
 - 2) копіювати виділений фрагмент в буфер обміну
 - 3) відмінити останню операцію
 - 4) вставити фрагмент з буферу обміну в активне вікно починаючи з позиції курсора
12. За допомогою команди **Cut** можна:
 - 1) знищити виділений фрагмент з занесенням його в буфер обміну
 - 2) копіювати виділений фрагмент в буфер обміну
 - 3) відмінити останню операцію
 - 4) вставити фрагмент з буферу обміну в активне вікно починаючи з позиції курсора
13. За допомогою команди **Copy** можна:
 - 1) знищити виділений фрагмент з занесенням його в буфер обміну
 - 2) копіювати виділений фрагмент в буфер обміну
 - 3) відмінити останню операцію
 - 4) вставити фрагмент з буферу обміну в активне вікно починаючи з позиції курсора
14. За допомогою команди **Paste** можна:
 - 1) знищити виділений фрагмент з занесенням його в буфер обміну
 - 2) копіювати виділений фрагмент в буфер обміну
 - 3) відмінити останню операцію
 - 4) знищити виділений фрагмент без занесення його в буфер обміну
15. За допомогою команди **Clear** можна:
 - 1) знищити виділений фрагмент без занесення його в буфер обміну
 - 2) знищити виділений фрагмент з занесенням його в буфер обміну
 - 3) відмінити останню операцію
 - 4) вставити фрагмент з буферу обміну в активне вікно починаючи з позиції курсора
16. За допомогою комбінації клавіш **Alt+BkSp** можна:
 - 1) знищити виділений фрагмент з занесенням його в буфер обміну
 - 2) копіювати виділений фрагмент в буфер обміну
 - 3) відмінити останню операцію
 - 4) вставити фрагмент з буферу обміну в активне вікно починаючи з позиції курсора
17. За допомогою комбінації клавіш **Shift+Del** можна:
 - 1) знищити виділений фрагмент з занесенням його в буфер обміну
 - 2) знищити виділений фрагмент без занесення його в буфер обміну
 - 3) відмінити останню операцію
 - 4) вставити фрагмент з буферу обміну в активне вікно починаючи з позиції курсора
18. За допомогою комбінації клавіш **Ctrl+Del** можна:
 - 1) знищити виділений фрагмент без занесення його в буфер обміну
 - 2) копіювати виділений фрагмент в буфер обміну
 - 3) знищити виділений фрагмент з занесенням його в буфер обміну
 - 4) вставити фрагмент з буферу обміну в активне вікно починаючи з позиції курсора
19. За допомогою комбінації клавіш **Ctrl+Ins** можна:

- 1) знищити виділений фрагмент з занесенням його в буфер обміну
 - 2) копіювати виділений фрагмент в буфер обміну
 - 3) відмінити останню операцію
 - 4) вставити фрагмент з буферу обміну в активне вікно починаючи з позиції курсора
20. За допомогою комбінації клавіш **Shift+Ins** можна:
- 1) знищити виділений фрагмент з занесенням його в буфер обміну
 - 2) копіювати виділений фрагмент в буфер обміну
 - 3) відмінити останню операцію
 - 4) вставити фрагмент з буферу обміну в активне вікно починаючи з позиції курсора
21. За допомогою команди **Run** можна виконати:
- 1) запуск програми на виконання з одночасним пошуком синтаксичних помилок
 - 2) запуск програми на виконання до позиції курсора
 - 3) покрокове виконання програми з входженням у всі функції
 - 4) покрокове виконання програми без входженням у всі функції (виконання функції як одна операція)
22. За допомогою команди **Go to cursor** можна виконати:
- 1) запуск програми на виконання з одночасним пошуком синтаксичних помилок
 - 2) запуск програми на виконання до позиції курсора
 - 3) покрокове виконання програми з входженням у всі функції
 - 4) покрокове виконання програми без входженням у всі функції (виконання функції як одна операція)
23. За допомогою команди **Trance into** можна виконати:
- 1) запуск програми на виконання з одночасним пошуком синтаксичних помилок
 - 2) запуск програми на виконання до позиції курсора
 - 3) покрокове виконання програми з входженням у всі функції
 - 4) покрокове виконання програми без входженням у всі функції (виконання функції як одна операція)
24. За допомогою команди **Step over** можна виконати:
- 1) запуск програми на виконання з одночасним пошуком синтаксичних помилок
 - 2) запуск програми на виконання до позиції курсора
 - 3) покрокове виконання програми з входженням у всі функції
 - 4) покрокове виконання програми без входженням у всі функції (виконання функції як одна операція)
25. За допомогою комбінації клавіш **Ctrl+F9** можна виконати:
- 1) запуск програми на виконання з одночасним пошуком синтаксичних помилок
 - 2) запуск програми на виконання до позиції курсора
 - 3) покрокове виконання програми з входженням у всі функції
 - 4) покрокове виконання програми без входженням у всі функції (виконання функції як одна операція)
26. За допомогою клавіші **F4** можна виконати:
- 1) запуск програми на виконання з одночасним пошуком синтаксичних помилок
 - 2) запуск програми на виконання до позиції курсора
 - 3) покрокове виконання програми з входженням у всі функції
 - 4) покрокове виконання програми без входженням у всі функції (виконання функції як одна операція)
27. За допомогою клавіші **F7** можна виконати:
- 1) запуск програми на виконання з одночасним пошуком синтаксичних помилок
 - 2) запуск програми на виконання до позиції курсора
 - 3) покрокове виконання програми з входженням у всі функції
 - 4) покрокове виконання програми без входженням у всі функції (виконання функції як одна операція)
28. За допомогою клавіші **F8** можна виконати:

- 1) запуск програми на виконання з одночасним пошуком синтаксичних помилок
 - 2) запуск програми на виконання до позиції курсора
 - 3) покрокове виконання програми з входженням у всі функції
 - 4) покрокове виконання програми без входженням у всі функції (виконання функції як одна операція)
29. За допомогою команди **Compile** можна виконати:
- 1) Компіляцію активної програми
 - 2) перевірку синтаксичних помилок в програмі
 - 3) додавання змінної до вікна watch для відслідковування її під час покрокового виконання програми
 - 4) знищити стрічку де знаходиться курсор
30. За допомогою команди **Make** можна виконати:
- 1) Компіляцію активної програми
 - 2) перевірку синтаксичних помилок в програмі
 - 3) додавання змінної до вікна watch для відслідковування її під час покрокового виконання програми
 - 4) знищити стрічку де знаходиться курсор
31. За допомогою команди **Add watch..** можна виконати:
- 1) Компіляцію активної програми
 - 2) перевірку синтаксичних помилок в програмі
 - 3) додавання змінної до вікна watch для відслідковування її під час покрокового виконання програми
 - 4) знищити стрічку де знаходиться курсор
32. За допомогою комбінації клавіш **Alt + F9** можна виконати:
- 1) Компіляцію активної програми
 - 2) перевірку синтаксичних помилок в програмі
 - 3) додавання змінної до вікна watch для відслідковування її під час покрокового виконання програми
 - 4) знищити стрічку де знаходиться курсор
33. За допомогою клавіші **F9** можна виконати:
- 1) Компіляцію активної програми
 - 2) перевірку синтаксичних помилок в програмі
 - 3) додавання змінної до вікна watch для відслідковування її під час покрокового виконання програми
 - 4) знищити стрічку де знаходиться курсор
34. За допомогою комбінації клавіш **Ctrl + F7** можна виконати:
- 1) Компіляцію активної програми
 - 2) перевірку синтаксичних помилок в програмі
 - 3) додавання змінної до вікна watch для відслідковування її під час покрокового виконання програми
 - 4) знищити стрічку де знаходиться курсор
35. За допомогою комбінації клавіш **Ctrl+Y** можна виконати:
- 1) Компіляцію активної програми
 - 2) перевірку синтаксичних помилок в програмі
 - 3) додавання змінної до вікна watch для відслідковування її під час покрокового виконання програми
 - 4) знищити стрічку де знаходиться курсор

СПИСОК ЛІТЕРАТУРИ ТА ДЖЕРЕЛ

0. Глинський Я.М., Анохін В.С., Ряжська В.А. Паскаль. Turbo Pascal і Delphi. Навч. посібн. 8-ме вид. – Львів: «СПД Глинський», 2007. – 192 с.
1. Жалдак М.І., Рамський Ю.С. Інформатика: навч. посібник/за ред.. М.І.Шкіля. – К.: Вища шк., 1991. – 319 с.
2. Спірін О.М., Кривонос О.М. Початки алгоритмізації та процедурного програмування: Методичний посібник для студ.вищих пед.навч.закл-ів фізико-математичних спеціальностей. – Житомир: ЖДПУ, 2002. – 93 с.
3. Присяжнюк А.В., Присяжнюк С.А. Вступ до програмування. Мова Паскаль: Популярний курс лекцій /Навчально-методичний посібник/. – Житомир: Поліграфічний центр ЖДПУ, 1999. – 182 с.
4. Сердюченко В.Я. Розробка алгоритмів та програмування мовою Turbo Pascal. – Харків: «Парітет», 1995. – 352 с.
5. www.e-olimp.com

Структурне програмування мовою Pascal (лабораторний практикум).

ДЛЯ НОТАТКІВ

Структурне програмування мовою Pascal (лабораторний практикум).

ДЛЯ НОТАТКІВ

Структурне програмування мовою Pascal (лабораторний практикум).

ДЛЯ НОТАТКІВ

Структурне програмування мовою Pascal (лабораторний практикум).

Навчально-методичне видання

ВАКАЛЮК Тетяна Анатоліївна
ЖУКОВСЬКИЙ Сергій Станіславович

Структурне програмування
мовою Pascal (лабораторний практикум)

Навчальний посібник для студентів фізико-математичного факультету

Надруковано з оригінал-макета авторів

Підписано до друку 30.05.11. Формат 60x90/16. Папір офсетний.

Гарнітура Times New Roman. Друк різнографічний.

Ум. друк. арк. 11. Обл. вид. арк. 10.5. Наклад 300. Зам. 745.

Видавець і виготовлювач

Видавництво Житомирського державного університету імені Івана Франка

м. Житомир, вул. Велика Бердичівська, 40

Свідоцтво суб'єкта видавничої справи:

серія ЖТ №10 від 07.12.04 р.

електронна пошта (E-mail): zu@zu.edu.ua