

Міністерство освіти і науки України
Національний університет водного господарства та природокористування
Навчально-науковий інститут автоматики, кібернетики та обчислювальної техніки
Кафедра прикладної математики

"ЗАТВЕРДЖУЮ"

Проректор з науково-педагогічної,
методичної та виховної роботи

_____ О.А. Лагоднюк

" ____ " _____ 2019 р.

04-01-20

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ
Program of the Discipline

Програмування
Programming

спеціальність	113 “Прикладна математика” 122 “Комп’ютерні науки”
specialty	113 “Applied mathematics” 122 “Computer Science”

Робоча програма «Програмування» для студентів, які навчаються за спеціальностями 113 «Прикладна математика», 122 «Комп'ютерні науки». Рівне: НУВГП, 2019. 16 с.

Розробники:

Остапчук Оксана Петрівна, к.т.н., доцент кафедри прикладної математики;
Іванчук Наталія Віталіївна, старший викладач кафедри прикладної математики.

Робочу програму схвалено на засіданні кафедри прикладної математики

Протокол від «28» серпня 2019 року № 16

Завідувач кафедри _____ П.М. Мартинюк

Схвалено науково-методичною комісією за спеціальністю 113 «Прикладна математика»

Протокол від «30» серпня 2019 року № 8

Голова науково-методичної комісії _____ П.М. Мартинюк

Схвалено науково-методичною комісією за спеціальністю 122 «Комп'ютерні науки»

Протокол від «30» серпня 2019 року № 6

Голова науково-методичної комісії _____ П.М. Мартинюк

© Остапчук О.П., 2019 рік
© Іванчук Н.В., 2019 рік
© НУВГП, 2019 рік

ВСТУП

Програма нормативної навчальної дисципліни «Програмування» складена відповідно до освітньо-професійної програми підготовки бакалавра спеціальностей «Прикладна математика» та «Комп'ютерні науки».

Міждисциплінарні зв'язки: дисципліна є складовою частиною блоку фундаментальної підготовки та відноситься до навчальних дисциплін циклу загальної підготовки студентів за спеціальностями «Прикладна математика» та «Комп'ютерні науки». Дисципліни, що вивчаються супутньо із зазначеною дисципліною: «Алгебра та геометрія», «Операційні системи», «Математичний аналіз», «Програмування і підтримка веб-застосувань» та є базою для вивчення таких дисциплін як «Рівняння математичної фізики», «Чисельні методи математичної фізики», «Математичне моделювання», «Методи обчислень» та ін.

Вимоги до знань та умінь визначаються галузевими стандартами вищої освіти України.

Анотація

Дисципліна «Програмування» спрямована на вивчення основ алгоритмізації, особливостей програмування інженерно-технічних та наукових задач мовою високого рівня, методів побудови та використання програмних комплексів, на опанування студентами практичних навиків програмного розв'язання математичних задач. Під час вивчення даної дисципліни студенти здобудуть знання, які допоможуть ефективно використовувати сучасні розробки технологій програмування при вивченні спеціальних дисциплін.

Ключові слова: алгоритмічна структура, вказівник, динамічна пам'ять, оператор, програма, типи даних, функція, C++, об'єктно-орієнтоване програмування.

Abstract

The discipline "Programming" is aimed at studying the fundamentals of algorithms, features of programming engineering and scientific problems in a high level language, methods of constructing and using software systems, and students mastering practical skills in software solving mathematical problems. During studying this discipline, students will gain knowledge that will help to effectively use modern programming technologies when learning special disciplines.

Keywords: algorithmic structure, pointer, dynamic memory, operator, program, data types, function, C++, object-oriented programming.

1. Опис навчальної дисципліни

Найменування показників	Галузь знань, напрям підготовки, професійна спрямованість	Характеристика навчальної дисципліни		
		денна форма навчання		
Кількість кредитів – 14	Галузь знань 11 Математика та статистика 12 Інформаційні технології	нормативна		
Модулів – 3	Спеціальність 113 Прикладна математика 122 Комп'ютерні науки	Рік підготовки:		
Змістових модулів – 5		1, 2-й		
Індивідуальне науково-дослідне завдання: –		Семестри		
Загальна кількість годин – 420		1-й	2-й	3-й
Тижневих годин для денної форми навчання: аудиторних – 4 самостійної роботи студента – 9,5	Рівень вищої освіти: бакалавр	Лекції – 90 год.		
		1-й семестр	2-й семестр	3-й семестр.
		26 год.	30 год.	34 год.
		Практичні, семінарські		
		-		
		Лабораторні – 90 год.		
		1-й семестр	2-й семестр	3-й семестр.
		26 год.	30 год.	34 год.
		Самостійна робота – 240 год.		
		1-й семестр	2-й семестр	3-й семестр.
		83 год.	75 год.	82 год.
		Індивідуальні завдання:		
		-		
Вид контролю				
1-й семестр	2-й семестр	3-й семестр.		
іспит	іспит	іспит		

Примітка.

Співвідношення кількості годин аудиторних занять до самостійної і індивідуальної роботи становить:
для денної форми навчання – 43% до 57%.

2. Мета та завдання навчальної дисципліни

Мета: засвоєння студентами основних понять і методів алгоритмізації, програмування та розв'язку інженерно-економічних задач з використанням персонального комп'ютера, мови програмування C++ та сучасних технологій програмування, формування наукового світогляду та здатності до засвоєння та постійного оновлення професійних знань, оскільки на сьогоднішній момент C++ - одна з найпотужніших і затребуваних мов програмування.

Завдання: сформувати у студентів знання методів алгоритмізації та розв'язання практичних задач з використанням сучасних технологій програмування; підготувати студентів до ефективного використання технологій програмування при вивченні спеціальних дисциплін, підготувати студентів до використання отриманих знань і навиків при розв'язуванні практичних задач, а також при написанні кваліфікаційних та магістерських робіт.

У результаті вивчення навчальної дисципліни студент повинен

знати:

1. Класичні і сучасні методи та прийоми програмування.
2. Базові засади ООП.
3. Поняття розумного вказівника.
4. Особливості перевантаження.
5. Принципи наслідування.

вміти:

- а) застосовувати прийоми та методи підготовки задач для їх розв'язання на ПК;
- б) вивчити основні засоби та прийоми алгоритмізації і програмування типових обчислювальних процесів;
- в) вивчити методи структурного та об'єктно-орієнтованого програмування;
- г) отримати практичні навички роботи в середовищах програмування C/C++;
- е) працювати з дружніми функціями;
- є) працювати з динамічними структурами даних;
- ж) використовувати потоки.

3. Програма навчальної дисципліни

МОДУЛЬ 1 (семестр 1)

ЗМІСТОВИЙ МОДУЛЬ 1

Реалізація типових алгоритмів за допомогою основних операторів та процедур мови C++

Вступ.

Предмет програмування, його роль у формуванні наукового світогляду, значення та можливості алгоритмізації та програмування при розв'язанні задач прикладної математики.

Тема 1. Основи алгоритмізації.

Основні етапи розв'язання задач на ЕОМ. Поняття про алгоритмізацію та алгоритм. Основні алгоритмічні структури.

Тема 2. Мови програмування, їх призначення та класифікація. Загальна характеристика мови програмування C++.

Історія створення C і C++. Найбільш відомі транслятори з C і C++. Етапи реалізації задачі на мовах C та C++. Алфавіт мови та ідентифікатори. Знаки операцій. Основні зарезервовані слова. Типи даних. Найпростіші конструкції мови. Основні стандартні функції. Арифметичні та логічні вирази та правила їх запису. Загальна структура C++-програми.

Тема 3. Основні оператори та процедури мови.

Оператор присвоєння. Введення та виведення даних. Програмування алгоритмів лінійної структури.

Тема 4. Програмування алгоритмів розгалуженої структури.

Оператор безумовного переходу. Умовний оператор *if*. Оператор вибору *switch*. Програмування алгоритмів розгалуженої структури.

Тема 5. Програмування алгоритмів циклічної структури.

Оператори циклу та реалізація циклічних алгоритмічних структур. Ітераційні цикли. Вкладені цикли. Алгоритмізація та програмування задач обчислення суми та добутку. Оператор продовження *continue*. Вихід з циклу. Оператор *break*.

Тема 6. Масиви даних.

Поняття масиву. Розміщення елементів масиву в пам'яті ПК. Класифікація масивів за вимірністю. Приклади масивів. Опис масивів в C++. Введення та виведення масивів. Використання двовимірних масивів. Типові алгоритми обробки масивів.

МОДУЛЬ 2 (семестр 2)

ЗМІСТОВИЙ МОДУЛЬ 2

Підпрограми. Обробка текстової інформації в C++

Тема 7. Підпрограми, їх організація та використання.

Доцільність введення підпрограм. Види підпрограм в C/C++. Підпрограми-функції. Формальні та фактичні параметри. Локальні та глобальні змінні і аргументи функцій. Значення параметрів за замовчуванням. Перевантаження функцій. Бібліотечні функції C++. Рекурсивні функції.

Тема 8. Обробка текстової інформації в C++.

Масиви символів. Стандартні функції для роботи зі стрічками. Функції перетворення типів. Вивід інформації в табличній формі.

ЗМІСТОВИЙ МОДУЛЬ 3

Робота з динамічними структурами даних

Тема 9. Вказівники та їх використання.

Означення вказівника. Опис вказівників. Вказівники і масиви. Динамічна пам'ять. Динамічні масиви. Посилання. Використання посилань при роботі з функціями.

Тема 10. Використання структур даних.

Поняття структури даних. Способи опису структури. Звертання до елементів структури та їх обробка. Ієрархічні (вкладені) структури. Об'єднання та перерахування.

Тема 11. Файли даних та їх обробка в C++.

Загальні відомості про файли. Класифікація файлів в C++. Текстові файли. Поняття текстового файлу та його опис. Особливості роботи з текстовими файлами в C++. Бінарні файли. Робота з бінарними файловими потоками у стилі C++. Довільний та послідовний доступ до компонент файлу.

Тема 12. Спискові структури даних.

Поняття списку. Робота зі стеком. Робота з чергою. Вставлення і вилучення елементів списку. Різновиди списків. Бінарні дерева.

Тема 13. Основні концепції об'єктно-орієнтованої методології програмування.

Базові поняття об'єктно-орієнтованого програмування. Основні концепції ООП. Класи і об'єкти в мові C++. Синтаксис опису класів та специфікатори доступу. «Друзі» класу. Конструктори, їх властивості. Різновиди конструкторів. Деструктори.

МОДУЛЬ 3 (семестр 3)

ЗМІСТОВИЙ МОДУЛЬ 4

Об'єктно-орієнтоване програмування

Тема 14. Конструктори та вказівники.

Початкова ініціалізація об'єктів класу. Перевантаження в ООП. Перевантажені конструктори. Вказівники на об'єкти. Вказівник this. Конструктор копіювання.

Тема 15. Перевантаження.

Константний метод. Перевантаження операторів. Перевантаження, які визначаються класом. Перевантаження унарних операцій.

Тема 16. Дружні функції.

Дружні функції. Дружні класи. Дружнє перевантаження. Статичні члени класу. Шаблон Сінглетон.

Тема 17. Поняття розумного вказівника.

Перевантаження оператора ->. Smart pointer. Функції з довільною кількістю і типом аргументів.

Тема 18. Динамічна структура даних.

Динамічні структури. Стек. Черга. Кільцева черга. Черга з пріоритетом (черга з пріоритетним виключенням, черга з пріоритетним включенням).

Тема 19. Список.

Списки в ООП. Однозв'язний список. Двозв'язний список. Шаблони класів.

Тема 20. Дерево.

Бінарне дерево. Структура дерева. Організація роботи (обхід дерева, пошук значення, отримання елементів, вставка елемента, видалення елемента).

Тема 21. Потоки.

Поняття потоку. Об'єкт-потік. Введення-виведення в мові C++. Файловий ввід/вивід із застосуванням потоків.

ЗМІСТОВИЙ МОДУЛЬ 5

Наслідування

Тема 22. Вкладені класи та наслідування.

Вкладений клас. Агрегування та композиція. Наслідування. Основні поняття наслідування. Специфікатори доступу при наслідуванні. Синтаксис організації наслідування. Множинне наслідування.

4. Структура навчальної дисципліни

Назви змістових модулів і тем	Кількість годин					
	Усього	денна форма				
		у тому числі				
		л	п	лаб.	інд.	с.р.
1	2	3	4	5	6	7
Модуль 1						
Змістовий модуль 1. Реалізація типових алгоритмів за допомогою основних операторів та процедур мови C++						
<i>Тема 1.</i> Вступ. Основи алгоритмізації.	15	2		-		13
<i>Тема 2.</i> Мови програмування, їх призначення та класифікація. Загальна характеристика мови програмування C++.	17	2		2		13
<i>Тема 3.</i> Основні оператори та процедури мови.	21	4		4		13
<i>Тема 4.</i> Програмування алгоритмів розгалуженої структури.	26	6		6		14
<i>Тема 5.</i> Програмування алгоритмів циклічної структури.	28	6		8		14
<i>Тема 6.</i> Масиви даних.	28	6		6		16
Разом за змістовим модулем 1	135	26		26		83
Разом за 1-й семестр	135	26		26		83
Модуль 2						
Змістовий модуль 2. Підпрограми. Обробка текстової інформації в C++						
<i>Тема 7.</i> Підпрограми, їх організація та використання.	13	4		4		5
<i>Тема 8.</i> Обробка текстової інформації в C++.	14	2		4		8
Разом за змістовим модулем 2	27	6		8		13
Змістовий модуль 3. Робота з динамічними структурами даних						
<i>Тема 9.</i> Вказівники та їх використання.	20	4		4		12
<i>Тема 10.</i> Використання структур даних.	20	4		4		12
<i>Тема 11.</i> Файли даних та їх обробка в C++.	22	6		4		12
<i>Тема 12.</i> Спискові структури даних.	26	6		6		14
<i>Тема 13.</i> Основні концепції об'єктно-орієнтованої методології програмування.	20	4		4		12
Разом за змістовим модулем 3	108	24		22		62
Всього за 2-й семестр	135	30		30		75

Модуль 3						
Змістовий модуль 4. Об'єктно-орієнтоване програмування						
<i>Тема 14.</i> Конструктори та вказівники.	12	2		2		8
<i>Тема 15.</i> Перевантаження.	14	4		2		8
<i>Тема 16.</i> Дружні функції.	14	4		2		8
<i>Тема 17.</i> Поняття розумного вказівника.	14	2		4		8
<i>Тема 18.</i> Динамічна структура даних.	18	4		4		10
<i>Тема 19.</i> Список.	19	4		4		11
<i>Тема 20.</i> Дерево.	14	4		4		6
<i>Тема 21.</i> Потоки.	25	6		6		13
Разом за змістовим модулем 4	130	30		28		72
Змістовий модуль 5. Наслідування						
<i>Тема 22.</i> Вкладені класи та наслідування.	20	4		6		10
Разом за змістовим модулем 5	20	4		6		10
Всього за 3-й семестр	150	34		34		82
Всього за курс	420	90		90		240

5. Теми лабораторних занять

№ з/п	Назва теми	Кількість годин
1	2	3
1.	Вирази. Оператори присвоювання. Стандартні функції.	2
2.	Лінійні алгоритми та програми.	2
3.	Алгоритми та програми розгалуженої структури.	4
4.	Циклічні алгоритми та програми.	4
5.	Типові прийоми програмування.	2
6.	Програмування ітераційних циклічних обчислювальних процесів.	4
7.	Одновимірні масиви.	4
8.	Багатовимірні масиви.	4
Всього за I-й семестр		26

9.	Обробка багатовимірних масивів.	2
10.	Програми з використанням підпрограм.	2
11.	Рекурсивні функції.	2
12.	Робота з текстовими змінними.	4
13.	Динамічні одновимірні масиви.	2
14.	Багатовимірні динамічні масиви.	2
15.	Використання структури.	4
16.	Файли даних. Робота з текстовими файлами.	4
17.	Динамічна структура даних (списки, стеки, черги).	4
18.	Використання ООП.	4
Всього за II-й семестр		30
19.	Оптимізація програми.	2
20.	Використання ООП.	4
21.	Перевантаження в ООП.	4
22.	«Дружба» в ООП.	4
23.	Робота з класом String.	4
24.	Шаблони класів.	4
25.	Динамічні структури даних.	6
26.	Потоки в C++.	6
Всього за III-й семестр		32
Всього за курс		90

6. Самостійна робота

Розподіл годин самостійної роботи для студентів включає наступні пункти:

1. Підготовка до аудиторних занять (0,5 год. на 1 год. аудиторних занять) – 90 год.
2. Підготовка до контрольних заходів (6 год. на 1 кредит) – 84 год.
3. Опрацювання окремих тем програми або їх частин, які не викладаються на лекціях – 66 год.

Розподіл навчального часу на вивчення дисципліни „Програмування”

6.1. Розподіл самостійної роботи студента

Число кредитів ЕСТС	Загальний обсяг дисципліни	Розподіл часу		Частка самостійної роботи, %
		Аудиторні заняття	Самостійна робота	
14	420	180	240	57
1-ий семестр				
4,5	135	52	83	61
2-ий семестр				
4,5	135	60	75	57
3-ий семестр				
5	150	68	82	55

1-ий семестр		
Опрацювання лекційного матеріалу	$0,5 \cdot 26 =$	13 годин
Підготовка до лабораторних робіт	$0,5 \cdot 26 =$	13 годин
Підготовка до екзамену	$6 \cdot 4,5 =$	27 годин
Всього		53 годин
Резерв		30 годин
2-ий семестр		
Опрацювання лекційного матеріалу	$0,5 \cdot 30 =$	15 годин
Підготовка до лабораторних робіт	$0,5 \cdot 30 =$	15 годин
Підготовка до екзамену	$6 \cdot 4,5 =$	27 годин
Всього		57 годин
Резерв		18 годин
3-ий семестр		
Опрацювання лекційного матеріалу	$0,5 \cdot 34 =$	17 годин
Підготовка до лабораторних робіт	$0,5 \cdot 34 =$	17 годин
Підготовка до екзамену	$6 \cdot 5 =$	30 годин
Всього		64 годин
Резерв		18 годин

6.2. Завдання для самостійної роботи 1-й семестр

№ з/п	Назва теми	К-ть год. сам. роботи
1.	Механізм використання багатовибірною розгалуження <i>switch</i> .	10
2.	Використання нескінченних циклів.	10
3.	Методи сортування масивів.	10
Загальна кількість годин		30

2-й семестр

№	Назва теми	К-ть год. сам.
---	------------	----------------

з/п		роботи
1.	Робота з бінарними файлами.	6
2.	Вказівники на функції.	6
3.	Дерева.	6
Загальна кількість годин		18

3-й семестр

№ з/п	Назва теми	К-ть год. сам. роботи
1.	Віртуальні функції. Поліморфізм.	4
2.	Виняткові ситуації.	4
3.	Стандартна бібліотека шаблонів (STL).	6
4.	Шаблони класів.	4
Загальна кількість годин		18

Звіт про самостійну роботу не надається. Вивчення відповідного матеріалу перевіряється під час поточного та підсумкового контролів знань.

7. Методи навчання

1. Лекції проводяться з використанням технічних засобів навчання і супроводжуються демонстрацією презентацій за допомогою відеопроєктора.
2. Лабораторні роботи проводяться в комп'ютерному класі з використанням роздаткового матеріалу та методичних вказівок.
3. Проведення контрольних тестувань.

8. Методи контролю

Оцінювання навчальних досягнень студентів за усіма видами навчальних робіт проводиться за *поточним* та *підсумковим* контролями. Поточний контроль знань студентів з навчальної дисципліни проводиться у письмовій формі та за допомогою тестів. Контрольні завдання включають тестові питання трьох рівнів складності.

Контроль самостійної роботи проводиться:

- з лекційного матеріалу – шляхом перевірки конспектів;
- з лабораторних робіт – за допомогою перевірки виконаних завдань та теоретичної підготовки до занять.

Усі контрольні заходи включено до 100-бальної шкали оцінювання.

Підсумковий семестровий контроль знань відбувається на екзамені у формі тестування. Основними критеріями, що характеризують рівень компетентності студента при оцінюванні результатів поточного та підсумкового контролів з навчальної дисципліни «Програмування» є:

- виконання всіх видів навчальної роботи, що передбачені робочою програмою навчальної дисципліни;
- глибина і характер знань навчального матеріалу за змістом навчальної дисципліни.

- характер відповідей на поставлені питання (чіткість, лаконічність, логічність, послідовність тощо);
- обґрунтування вибору методу для розв'язання задач;
- рівень вміння аналізувати одержані результати.

Оцінювання результатів усіх форм контролю передбачено у 100-бальній шкалі.

Критерії оцінювання результатів поточної роботи (завдань, що виконуються на практичних заняттях, результати самостійної роботи студентів) проводиться у % від кількості балів, виділених на завдання, із заокругленням до цілого числа:

0% – завдання не виконано;

40% – завдання виконано частково та містить суттєві помилки методичного або розрахункового характеру;

60% – завдання виконано повністю, але містить суттєві помилки у розрахунках або в методиці;

80% – завдання виконано повністю і вчасно, проте містить окремі несуттєві недоліки (розмірності, висновки, оформлення тощо);

100% – завдання виконано правильно, вчасно і без зауважень.

9. Розподіл балів, які отримують студенти

1-й семестр

Модуль 1. Поточне тестування та СРС						Підсумковий контроль	Сума
Змістовий модуль 1						40	100
60							
T1	T2	T3	T4	T5	T6		
5	5	10	10	10	10		

T1, T2, ..., T6 – теми змістових модулів

2-й семестр

Модуль 2. Поточне тестування та СРС							Підсумковий контроль	Сума
Змістовий модуль 2		Змістовий модуль 3					40	100
16		44						
T7	T8	T9	T10	T12	T12	T13		
8	8	8	8	9	9	10		

T7, T8, ..., T13 – теми змістових модулів

3-й семестр

Модуль 3. Поточне тестування та СРС								Підсумков. контроль	Загальна кількість балів	
Змістовий модуль 4							Змістовий модуль 5	40	100	
52							8			
T14	T15	T16	T17	T18	T19	T20	T21			T22
6	8	6	6	6	6	6	8			8

T14, T15...T22 – теми змістових модулів.

Шкала оцінювання

Сума балів за всі види навчальної діяльності	Оцінка за національною шкалою
	для екзамену
90–100	відмінно
82–89	добре
74–81	
64–73	задовільно
60–63	
35–59	незадовільно з можливістю повторного складання
0–34	незадовільно з обов'язковим повторним вивченням дисципліни

10. Методичне забезпечення

Методичне забезпечення навчальної дисципліни «Програмування» включає:

1. Опорний конспект лекцій (у електронному і паперовому вигляді) за всіма темами курсу, у тому числі і для самостійного вивчення.

2. Пакети тестових завдань по кожній темі і в цілому по всьому курсу дисципліни (навчальна платформа Moodle).

3. 04-01-33, 04-01-34 Остапчук О.П., Роценюк А.М. Методичні вказівки для виконання лабораторних робіт з дисципліни «програмування» студентами спеціальностей 113 «Прикладна математика» і 122 «Комп'ютерні науки» першого (бакалаврського) рівня вищої освіти денної та заочної форм навчання (електронний варіант), 2019 р.

11. Рекомендована література

Базова література

1. Давыдов В.Г. Технологии программирования С++. СПб.: БХВ, 2005. 672 с.
2. Прата С. Язык программирования С++: Лекции и упражнения. Москва: ДиаСофтЮП, 2005. 1104 с.
3. Программирование на С++: Учебное пособие/ Под. ред. А.Д.Хомоненко. СПб.: КОРОНА-ПРИНТ, 2003. 512 с.
4. Трофименко О.Г., Прокоп Ю.В., Швайко І.Г., Буката Л.М. та ін. С++. Основи програмування. Теорія та практика: підручник /за ред. О.Г.Трофименко. Одеса: Фенікс, 2010. 544 с.
5. Шилдт Г. Самоучитель С++: Пер. с англ. - 3-е изд. СПб.: БХВ-Петербург, 2005. 688 с.
6. Элджер Д. Библиотека программиста С++. М.: ООО «И.Д. Вильямс», 2008. 1376 с.
7. Александреску А. Современное программирование на С++. М.: ООО «И.Д. Вильямс», 2008. 336 с.

Допоміжна

1. Власюк А.П., Мартинюк П.М., Прищепа О.В., Філатова І.А., Філатов М.С., Рощенюк А.М., Демчук О.С., Демчук М.Б., Мічута О.Р., Цветкова Т.П., Федорчук Н.А. Лабораторний практикум з програмування. Навч. посібник / за ред. А.П.Власюка. Рівне: НУВГП, 2010. 495 с.
2. Ишкова Э.А. С++. Начала программирования. Москва: БИНОМ, 2004. 368 с.
3. Лафоре Р. Объектно-ориентированное программирование в С++. СПб.: Питер, 2003. 928 с.
4. Павловская Т.А. Программирование на языке высокого уровня С/С++. СПб.: Питер, 2002. 464 с.
5. Павловская Т.А., Щупак Ю.А. С/С++. Структурное программирование: Практикум. СПб.: Питер, 2003. 240 с.
6. Пильщиков В.Н., Горячая И.В., Бордаченкова Е.А. Решение задач с использованием рекурсии. Учебно-методическое пособие. Москва. 2012. 37 с.
7. Пономарёв В. Программирование на С++/С# в Visual Studio .NET 2003. СПб.: БХВ, 2004. 352 с.

12. Інформаційні ресурси

1. Рівненська обласна універсальна бібліотека. URL: <http://libr.rv.ua/>
2. Наукова бібліотека НУВГП. URL: <http://lib.nuwm.edu.ua/>
3. Національна бібліотека ім. В.І. Вернадського URL: <http://www.nbuv.gov.ua/>
4. Metanit.com. URL: <http://metanit.com/>