

**КИЇВСЬКИЙ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ  
ІМЕНІ ТАРАСА ШЕВЧЕНКА  
МЕХАНІКО-МАТЕМАТИЧНИЙ ФАКУЛЬТЕТ**

**ІНФОРМАТИКА ТА ПРОГРАМУВАННЯ**

**РОБОЧА НАВЧАЛЬНА ПРОГРАМА ДИСЦИПЛІНИ**

**КИЇВ – 2015**

**Київський національний університет  
імені Тараса Шевченка  
Механіко-математичний факультет  
Кафедра математичної фізики**

"Затверджую"

Голова Вченої ради \_\_\_\_\_ М.Ф. Городній

**ІНФОРМАТИКА ТА ПРОГРАМУВАННЯ**

**РОБОЧА НАВЧАЛЬНА ПРОГРАМА ДИСЦИПЛІНИ**  
освітньо-професійних програм підготовки бакалаврів математики (6.040201)  
та статистики (6.040205)

I курс

Затверджено  
Вченою радою  
механіко-математичного факультету  
протокол № \_\_\_\_ від \_\_\_\_\_.\_\_\_\_.2015 р.

Погоджено з науково-методичною комісією  
\_\_\_\_\_.\_\_\_\_\_.2015 р.

\_\_\_\_\_  
Підпис голови НМК ф-ту

Викладачі

к.т.н., доцент Обвінцев О.В.  
к.ф.м.н., доцент Крєневич А.П.  
к.ф.м.н., доцент Довгий Б.П.

**КИЇВ-2015**

## Методичні рекомендації з вивчення дисципліни

Дисципліна "Інформатика та програмування" є базовою дисципліною для всіх спеціальностей механіко-математичного факультету, яка викладається у першому та другому семестрах в обсязі 4-х кредитів (за Європейською Кредитно-Трансферною Системою ECTS) в тому числі 104 годин аудиторних занять, з них 52 години лекцій, 52 години лабораторних занять, 22 годин самостійної роботи і закінчується іспитом у першому семестрі та заліком другому семестрі.

**Метою і завданням навчальної дисципліни** "Інформатика та програмування" є: ознайомлення та оволодіння сучасними методами та теоретичними положеннями, притаманними інформатиці та програмній інженерії, та їх застосування при побудові алгоритмів та програм для сучасних комп'ютерів.

**Предмет навчальної дисципліни** "Інформатика та програмування" включає основні методи та поняття інформатики – виконавець, алгоритм, програма, лінійні, розгалужені та циклічні алгоритми та програми, їх побудова, основні властивості програм, типи даних, підпрограми, модулі, класи та об'єкти.

### **Вимоги до знань та вмінь.**

*Знати:* основні поняття інформатики - виконавець, алгоритм, програма; синтаксис, семантику та властивості структур керування (ланцюг, розгалуження, цикли); формальні методи побудови алгоритмів та програм за допомогою рекурентних співвідношень, будову простих та складених типів даних; опис та використання підпрограм; синтаксичні конструкції мови програмування Python.

*Вміти:* будувати лінійні, розгалужені та циклічні алгоритми та програми, які оперують даними простих та складених типів, з використанням підпрограм, у мові програмування Python.

**Система контролю знань та умови складання заліку.** Навчальна дисципліна оцінюється за модульно-рейтинговою системою. Вона складається з чотирьох модулів: до першого входять 1, 2, 3 та 4 теми, до другого – 4, 5, 6, 7, 8, 9, 10 теми, до третього – 11, 12, 13, 14, 15 теми, до четвертого – 16, 17, 18 та 19 теми.

Результати навчальної діяльності студентів оцінюються за 100-бальною шкалою у кожному з семестрів.

**Форми поточного контролю:** оцінювання домашніх самостійних завдань; тестів та контрольних робіт, виконаних студентами під час практичних занять. Студент може отримати максимально 20 балів за усні відповіді та доповнення на практичних заняттях в змістових модулях 1, 2, 3 та 4.

**Модульний контроль:** колоквиум та модульні контрольні роботи.

## ТЕМАТИЧНИЙ ПЛАН ДИСЦИПЛІНИ

№ теми	Назва теми	Кількість годин		
		Лекції	Лабораторні заняття	Самостійна робота
<b>МОДУЛЬ 1. СТРУКТУРИ КЕРУВАННЯ</b>				
1.	Лінійні програми	2	2	2
2.	Розгалужені програми	2	2	2
3.	Циклічні програми	6	6	2
4.	Прості типи даних	4	4	2
<b>МОДУЛЬ 2. СТРУКТУРИ ДАНИХ</b>				
5.	Символи та рядки	2	2	2
6.	Списки	2	2	2
7.	Кортежі	2	2	2
8.	Словники	2	2	2
9.	Підпрограми	6	6	2
10.	Модулі та пакети	2	2	2
<b>МОДУЛЬ 3. ОБ'ЄКТНО-ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ</b>				
11.	Множини	2	2	2
12.	Файли	2	2	2
13.	Класи та об'єкти	2	2	2
14.	Рекурсивні структури даних	4	4	2
15.	Обробка помилок та виключних ситуацій	2	2	2
<b>МОДУЛЬ 4. МЕТАПРОГРАМУВАННЯ</b>				
16.	Ітератори та генератори	2	2	2
17.	Декоратори	2	2	2
18.	Множинне наслідування	2	2	2
19.	Метакласи та метапрограмування	4	4	2
Всього годин за рік 142, з них		52	52	38

В результаті виконання самостійних робіт, домашніх завдань, опитування студентів на лекціях та практичних заняттях, перевірки знань студентів під час розв'язання задач біля дошки, виконання модульної контрольної роботи протягом кожного семестру студент може отримати 60 балів:

Семестр	Модуль	Лабораторні заняття	Модульний контроль	Всього
I	Модуль 1	20	10	30
I	Модуль 2	20	10	30
I	Всього			60
II	Модуль 3	20	10	30
II	Модуль 4	20	10	30
II	Всього			60

На лабораторних заняттях студент може отримати від 1 до 3 балів за кожен правильну відповідь (розв'язок задачі) біля дошки або за комп'ютером. Повне та якісне виконання домашнього завдання може оцінюватись 1 балом. За виконання поточних контрольних робіт під час лабораторних занять студент може отримати до 10 балів.

Для допуску до заліку (іспиту) студент повинен отримати протягом семестру не менше 35 балів.

На заліку студент може отримати максимально 40 балів.

Якщо студент набрав протягом семестру більше, ніж 45 балів, він може відмовитись від складання заліку. У цьому випадку до його рейтингу у семестрі додаються додаткові бали без складання заліку в кількості, що залежить від набраних балів у семестрі:

від 46 до 50 балів – додається 14 балів

від 51 до 55 балів – додається 24 балів

від 56 до 60 балів – додається 34 бали

Набравши протягом семестру більше, ніж 45 балів, студент може підвищити свій рейтинг в разі успішної здачі заліку.

Після складання іспиту студенти отримують "задовільно", якщо набирають протягом семестру і на заліку разом 60 - 74 бали, «добре», якщо 75 – 89 балів, «відмінно», якщо 90 – 100 балів.

Студенті, які протягом семестру і на заліку набрали менше 60 балів, отримують "незадовільно".

Після складання заліку студенти отримують "зараховано", якщо набирають протягом семестру і на заліку разом не менше 60 балів.

Студенті, які протягом семестру і на заліку набрали менше 60 балів, отримують "не зараховано".

# ТЕМИ ЛЕКЦІЙ, ПРАКТИЧНИХ ЗАНЯТЬ ТА ЗАВДАННЯ ДЛЯ САМОСТІЙНОЇ РОБОТИ

## І семестр

### ЗМІСТОВИЙ МОДУЛЬ 1. СТРУКТУРИ КЕРУВАННЯ.

#### ТЕМА № 1. Лінійні програми

**Лекція 1.** Лінійні програми – 2 год.

Коротка історія розвитку обчислювальної техніки. Визначення інформатики та інформації. Поняття алгоритму та виконавця. Програми та програмування, компілятори та інтерпретатори. Основні команди Python. Ланцюги та лінійні програми. Рівносильність інструкцій. Правила запису програм.

**Лабораторне заняття № 1.** Лінійні програми – 2 год.

**Самостійна робота з вивчення матеріалу лекції № 1** – 2 год.

#### Література

1. A Byte of Python (Russian) Версія 2.01 Swaroop С Н (Translated by Vladimir Smolyar),  
<http://wombat.org.ua/AByteOfPython/AByteofPythonRussian-2.01.pdf>
2. Бублик В.В., Личман В.В., Обвінцев О.В.. Інформатика та програмування. Електронний конспект лекцій, 2003 р.,
3. Марк Лутц, Изучаем Python, 4-е издание, 2010, Символ-Плюс
4. Самоучитель Python. <http://pythonworld.ru/samouchitel-python>
5. Python 3.4.3 documentation
6. С. Шапошникова. Основы программирования на Python. Версия 2 (2011). <http://younglinux.info/pdf>

#### ТЕМА № 2. Розгалужені програми.

**Лекція 2.** Розгалужені програми. – 2 год.

Алгебра висловлювань, бульові операції, їх властивості. Відношення та їх властивості. Умови. Розгалуження (звичайне розгалуження, захищена команда, каскадне розгалуження). Розгалужені програми.

**Лабораторне заняття № 2.** Побудова розгалужених програм. – 2 год.

**Самостійна робота з вивчення матеріалу лекції № 2 – 2 год.**

**Література**

1. A Byte of Python (Russian) Версія 2.01 Swaroop С Н (Translated by Vladimir Smolyar),  
<http://wombat.org.ua/AByteOfPython/AByteofPythonRussian-2.01.pdf>
2. Бублик В.В., Личман В.В., Обвінцев О.В.. Інформатика та програмування. Електронний конспект лекцій, 2003 р.,
3. Марк Лутц, Изучаем Python, 4-е издание, 2010, Символ-Плюс
4. Самоучитель Python. <http://pythonworld.ru/samouchitel-python>
5. Python 3.4.3 documentation
6. С. Шапошникова. Основы программирования на Python. Версия 2 (2011). <http://younglinux.info/pdf>

**ТЕМА № 3. Циклічні програми.**

**Лекція № 3.** Цикл з умовою продовження – 2 год.

Поняття циклу та циклічної програми. Цикл з умовою продовження, його властивості.

**Самостійна робота з вивчення матеріалу лекції № 3 – 2 год.**

**Лабораторне заняття № 3.** Цикл з умовою продовження. – 2 год.

**Лекція 4.** Рекурентні співвідношення.

Рекурентні співвідношення 1 та вищих порядків, системи рекурентних співвідношень. Правила обчислення елементів послідовностей, заданих рекурентними співвідношеннями.

**Лабораторне заняття № 4.** Рекурентні співвідношення. – 2 год.

**Лекція 5.** Рекурентні обчислення за умовою. Цикл по діапазону значень.

Повний синтаксис циклу за умовою, обчислення границь. Цикл по діапазону значень.

**Самостійна робота з вивчення матеріалу лекції № 4 – 2 год.**

**Лабораторне заняття № 5.** Рекурентні обчислення за умовою. Цикл по діапазону значень. – 2 год.

**Самостійна робота з вивчення матеріалу лекції № 5** – 2 год.

**Література**

1. A Byte of Python (Russian) Версія 2.01 Swaroop С Н (Translated by Vladimir Smolyar),  
<http://wombat.org.ua/AByteOfPython/AByteofPythonRussian-2.01.pdf>
2. Бублик В.В., Личман В.В., Обвінцев О.В.. Інформатика та програмування. Електронний конспект лекцій, 2003 р.,
3. Марк Лутц, Изучаем Python, 4-е издание, 2010, Символ-Плюс
4. Самоучитель Python. <http://pythonworld.ru/samouchitel-python>
5. Python 3.4.3 documentation
6. С. Шапошникова. Основы программирования на Python. Версия 2 (2011). <http://younglinux.info/pdf>

**ТЕМА № 4. Числові типи даних**

**Лекція 6.** Арифметика наближених обчислень – 2 год.

Поняття типу даних. Складові частини типу даних. Позиційний запис дійсного числа. Функція округлення, похибка округлення. Наближені операції та їх властивості

**Лабораторне заняття № 6.** Арифметика наближених обчислень – 2 год.

**Лекція 7.** Числові типи даних – 2 год.

Цілий тип даних. Дійсний тип даних. Комплексний тип даних

**Лабораторне заняття № 7.** Числові типи даних – 2 год.

**Самостійна робота з вивчення матеріалу лекції № 6 - 7** – 4 год.

**Література**

1. A Byte of Python (Russian) Версія 2.01 Swaroop С Н (Translated by Vladimir Smolyar),  
<http://wombat.org.ua/AByteOfPython/AByteofPythonRussian-2.01.pdf>
2. Бублик В.В., Личман В.В., Обвінцев О.В.. Інформатика та програмування. Електронний конспект лекцій, 2003 р.,
3. Марк Лутц, Изучаем Python, 4-е издание, 2010, Символ-Плюс
4. Python 3.4.3 documentation



5. <http://deeplearning.net/software/theano/tutorial/python-memory-management.html>
6. <http://www.laurentluce.com/posts/python-integer-objects-implementation/>
7. <https://docs.python.org/2/tutorial/float.html>
8. <http://www.johndcook.com/blog/2009/04/06/anatomy-of-a-floating-point-number/>

## ЗМІСТОВИЙ МОДУЛЬ 2. СТРУКТУРИ ДАНИХ.

### ТЕМА № 5. Символи та рядки.

**Лекція 8.** Символи та рядки. – 2 год.

Символи. Стандарти кодування символів. Unicode. Кодування символів у Unicode та формати представлення символів. Рядки: носій, операції, відношення та інструкції. Рядки як послідовності. Вирізки та їх використання. Додаткові функції для рядків.

**Лабораторне заняття № 8.** Тип даних рядок – 2 год.

**Самостійна робота з вивчення матеріалу лекції № 8**– 2 год.

#### Література

1. A Byte of Python (Russian) Версія 2.01 Swaroop С Н (Translated by Vladimir Smolyar),  
<http://wombat.org.ua/AByteOfPython/AByteofPythonRussian-2.01.pdf>
2. Бублик В.В., Личман В.В., Обвінцев О.В.. Інформатика та програмування. Електронний конспект лекцій, 2003 р.,
3. Марк Лутц, Изучаем Python, 4-е издание, 2010, Символ-Плюс
4. Python 3.4.3 documentation
5. <https://docs.python.org/3/howto/unicode.html>
6. <http://unicode-table.com/ru/>
7. <http://www.joelonsoftware.com/articles/Unicode.html> (переклад російською  
[http://local.joelonsoftware.com/wiki/%D0%90%D0%B1%D1%81%D0%BE%D0%BB%D1%8E%D1%82%D0%BD%D1%8B%D0%B9\\_%D0%9C%D0%B8%D0%BD%D0%B8%D0%BC%D1%83%D0%BC,%D0%BA%D0%BE%D1%82%D0%BE%D1%80%D1%8B%D0%B9\\_%D0%9A%D0%B0%D0%B6%D0%B4%D1%8B%D0%B9\\_%D0%A0%D0%B0%D0%B7%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D1%87%D0%B8%D0%BA](http://local.joelonsoftware.com/wiki/%D0%90%D0%B1%D1%81%D0%BE%D0%BB%D1%8E%D1%82%D0%BD%D1%8B%D0%B9_%D0%9C%D0%B8%D0%BD%D0%B8%D0%BC%D1%83%D0%BC,%D0%BA%D0%BE%D1%82%D0%BE%D1%80%D1%8B%D0%B9_%D0%9A%D0%B0%D0%B6%D0%B4%D1%8B%D0%B9_%D0%A0%D0%B0%D0%B7%D1%80%D0%B0%D0%B1%D0%BE%D1%82%D1%87%D0%B8%D0%BA)

[%D0%9F%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%BD%D0%BE%D0%B3%D0%BE\\_%D0%9E%D0%B1%D0%B5%D1%81%D0%BF%D0%B5%D1%87%D0%B5%D0%BD%D0%B8%D1%8F\\_%D0%9E%D0%B1%D1%8F%D0%B7%D0%B0%D1%82%D0%B5%D0%BB%D1%8C%D0%BD%D0%BE\\_%D0%94%D0%BE%D0%BB%D0%B6%D0%B5%D0%BD\\_%D0%97%D0%BD%D0%B0%D1%82%D1%8C\\_%D0%BE\\_Unicode\\_%D0%B8\\_%D0%9D%D0%B0%D0%B1%D0%BE%D1%80%D0%B0%D1%85\\_%D0%A1%D0%B8%D0%BC%D0%B2%D0%BE%D0%BB%D0%BE%D0%B2\)](https://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%BD%D0%BE%D0%B3%D0%BE_%D0%9E%D0%B1%D0%B5%D1%81%D0%BF%D0%B5%D1%87%D0%B5%D0%BD%D0%B8%D1%8F_%D0%9E%D0%B1%D1%8F%D0%B7%D0%B0%D1%82%D0%B5%D0%BB%D1%8C%D0%BD%D0%BE_%D0%94%D0%BE%D0%BB%D0%B6%D0%B5%D0%BD_%D0%97%D0%BD%D0%B0%D1%82%D1%8C_%D0%BE_Unicode_%D0%B8_%D0%9D%D0%B0%D0%B1%D0%BE%D1%80%D0%B0%D1%85_%D0%A1%D0%B8%D0%BC%D0%B2%D0%BE%D0%BB%D0%BE%D0%B2))

8. [%D0%AE%D0%BD%D0%B8%D0%BA%D0%BE%D0%B4](https://ru.wikipedia.org/wiki/%D0%AE%D0%BD%D0%B8%D0%BA%D0%BE%D0%B4)

## **ТЕМА № 6. Списки.**

**Лекція 9.** Списки. – 2 год.

Списки: носій, операції, відношення та інструкції для списків. Реалізація масивів на базі списків. Взаємозв'язок між рядками та списками. Спискоутворення. Функція enumerate

**Лабораторне заняття № 9.** Списки – 2 год.

**Самостійна робота з вивчення матеріалу лекції № 9**– 2 год.

### **Література**

1. A Byte of Python (Russian) Версия 2.01 Swaroop С Н (Translated by Vladimir Smolyar),  
<http://wombat.org.ua/AByteOfPython/AByteofPythonRussian-2.01.pdf>
2. Марк Лутц, Изучаем Python, 4-е издание, 2010, Символ-Плюс
3. Python 3.4.3 documentation
4. Бублик В.В., Личман В.В., Обвінцев О.В.. Інформатика та програмування. Електронний конспект лекцій, 2003 р.,

## **ТЕМА № 7. Кортежі.**

**Лекція 10.** Кортежі. – 2 год.

Кортежі. Носій для кортежів. Операції, відношення та інструкції для кортежів. Пакування та розпакування. Функції zip та map. Іменовані кортежі.

**Лабораторне заняття № 10.** Кортежі – 2 год.

**Самостійна робота з вивчення матеріалу лекції № 10**– 2 год.

### **Література**

1. A Byte of Python (Russian) Версія 2.01 Swaroop С Н (Translated by Vladimir Smolyar),  
<http://wombat.org.ua/AByteOfPython/AByteofPythonRussian-2.01.pdf>
2. Марк Лутц, Изучаем Python, 4-е издание, 2010, Символ-Плюс
3. Python 3.4.3 documentation
4. Бублик В.В., Личман В.В., Обвінцев О.В.. Інформатика та програмування. Електронний конспект лекцій, 2003 р.,

## **ТЕМА № 8. Словники.**

**Лекція 11.** Словники. – 2 год.

Словники. Носій для словників. Операції, відношення та інструкції для словників. Функції all any та filter. Словникоутворення.

**Лабораторне заняття № 11.** Словники – 2 год.

**Самостійна робота з вивчення матеріалу лекції № 11**– 2 год.

### **Література**

1. A Byte of Python (Russian) Версія 2.01 Swaroop С Н (Translated by Vladimir Smolyar),  
<http://wombat.org.ua/AByteOfPython/AByteofPythonRussian-2.01.pdf>
2. Марк Лутц, Изучаем Python, 4-е издание, 2010, Символ-Плюс
3. Python 3.4.3 documentation
4. Марк Саммерфилд, Программирование на Python 3. Подробное руководство. - Символ-Плюс, 2009.
5. [http://www.python-course.eu/python3\\_dictionaries.php](http://www.python-course.eu/python3_dictionaries.php)

## **ТЕМА № 9. Підпрограми.**

**Лекція 12.** Функції – 2 год.

Означення підпрограми. Простий синтаксис та просте правило виклику функції. Передача параметрів різних типів у підпрограми. Функція format. Оформлення документації функції. Повернення функцією декількох результатів. Значення параметрів за угодою. Позиційні та ключові параметри. Змінна кількість параметрів функції. Повний синтаксис та повне правило виклику функції

**Лабораторне заняття № 12.** Функції 2 год.

**Лекція 13.** Рекурсія. – 2 год.

lambda-функції. Рекурсія та рекурсивні підпрограми

**Лабораторне заняття № 13.** Рекурсія. – 2 год.

**Лекція 14.** Функціональний тип даних. – 2 год.  
Глобальні змінні. Функціональний тип даних

**Лабораторне заняття № 14.** Функціональний тип даних. – 2 год.

**Самостійна робота з вивчення матеріалу лекцій № 12-14 - 2 год.**

#### **Література**

1. A Byte of Python (Russian) Версия 2.01 Swaroop С Н (Translated by Vladimir Smolyar),  
<http://wombat.org.ua/AByteOfPython/AByteofPythonRussian-2.01.pdf>
2. Бублик В.В., Личман В.В., Обвінцев О.В.. Інформатика та програмування. Електронний конспект лекцій, 2003 р.,
3. Марк Лутц, ИзучаемPython, 4-е издание, 2010, Символ-Плюс
4. Python 3.4.3 documentation
5. Марк Саммерфилд, Программирование на Python 3. Подробное руководство. - Символ-Плюс, 2009.
6. [http://www.python-course.eu/python3\\_functions.php](http://www.python-course.eu/python3_functions.php)

#### **ТЕМА № 10. Модулі та пакети.**

**Лекція 15.** Модулі та пакети. – 2 год.

Означення модуля. Модулі у Python. Імпорт модулів та використання об'єктів модулів. Видимість об'єктів модуля. Области дії імен. Пакети. Стандартні модулі. Рядки документації модуля.

**Лабораторне заняття № 15.** Модулі та пакети. – 2 год.

**Самостійна робота з вивчення матеріалу лекції № 15– 2 год.**

#### **Література**

1. A Byte of Python (Russian) Версия 2.01 Swaroop С Н (Translated by Vladimir Smolyar),  
<http://wombat.org.ua/AByteOfPython/AByteofPythonRussian-2.01.pdf>
2. Бублик В.В., Личман В.В., Обвінцев О.В.. Інформатика та програмування. Електронний конспект лекцій, 2003 р.,
3. Марк Лутц, ИзучаемPython, 4-е издание, 2010, Символ-Плюс
4. Python 3.4.3 documentation
5. Марк Саммерфилд, Программирование на Python 3. Подробное руководство. - Символ-Плюс, 2009.

## **ЗМІСТОВИЙ МОДУЛЬ 3. ОБ'ЄКТНО-ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ.**

### **ТЕМА № 11. Множини.**

**Лекція 16.** Множини. – 2 год.

Множини: носій, основні операції, відношення та інструкції для множин. Додаткові операції, відношення та інструкції для множин. Множини, що не змінюються. Множиноутворення.

**Лабораторне заняття № 16.** Множини. – 2 год.

**Самостійна робота з вивчення матеріалу лекції № 16**– 2 год.

#### **Література**

1. A Byte of Python (Russian) Версия 2.01 Swaroop С Н (Translated by Vladimir Smolyar),  
<http://wombat.org.ua/AByteOfPython/AByteofPythonRussian-2.01.pdf>
2. Марк Лутц, ИзучаемPython, 4-е издание, 2010, Символ-Плюс
3. Python 3.4.3 documentation
4. Бублик В.В., Личман В.В., Обвінцев О.В.. Інформатика та програмування. Електронний конспект лекцій, 2003 р.,
5. [http://www.python-course.eu/python3\\_sets\\_frozensets.php](http://www.python-course.eu/python3_sets_frozensets.php)

### **ТЕМА № 12. Файли.**

**Лекція 17.** Файли – 2 год.

Файли, носій типу файл, текстові та нетекстові файли. Відкриття та закриття файлів. Читання з файлу та запис у файл. Додаткові дії для файлів. Запис та читання з текстового файлу нетекстових значень. Параметри при запуску програми з командного рядка. Обробка нетекстових файлів:pickle та shelve.

**Лабораторне заняття № 17.** Файли – 2 год.

**Самостійна робота з вивчення матеріалу лекції № 17**– 2 год.

#### **Література**

1. A Byte of Python (Russian) Версия 2.01 Swaroop С Н (Translated by Vladimir Smolyar),  
<http://wombat.org.ua/AByteOfPython/AByteofPythonRussian-2.01.pdf>
2. Марк Лутц, ИзучаемPython, 4-е издание, 2010, Символ-Плюс
3. Python 3.4.3 documentation

4. Бублик В.В., Личман В.В., Обвінцев О.В.. Інформатика та програмування. Електронний конспект лекцій, 2003 р.,
5. [http://www.python-course.eu/python3\\_file\\_management.php](http://www.python-course.eu/python3_file_management.php)

### **ТЕМА № 13. Класи та об'єкти.**

**Лекція 18.** Класи та об'єкти – 2 год.

Об'єктно-орієнтований підхід до програмування. Визначення класів та об'єктів, інші поняття ООП. Синтаксис опису класів у Python. Створення та використання об'єктів класу. Зв'язування між об'єктами та методами. Віртуальні методи. Обмеження доступу до полів та методів класу. Рядки документації класу. Графічна бібліотека turtle. Статичні поля та статичні методи класів. Модуль для роботи з псевдовипадковими величинами. Збереження змінних у пам'яті. Визначення типу та відношення для об'єктів

**Лабораторне заняття № 18.** Класи та об'єкти – 2 год.

**Самостійна робота з вивчення матеріалу лекції № 18**– 2 год.

#### **Література**

1. A Byte of Python (Russian) Версія 2.01 Swaroop С Н (Translated by Vladimir Smolyar),  
<http://wombat.org.ua/AByteOfPython/AByteofPythonRussian-2.01.pdf>
2. Бублик В.В., Личман В.В., Обвінцев О.В.. Інформатика та програмування. Електронний конспект лекцій, 2003 р.,
3. Марк Лутц, ИзучаемPython, 4-е издание, 2010, Символ-Плюс
4. Python 3.4.3 documentation
5. Марк Саммерфилд, Программирование на Python 3. Подробное руководство. - Символ-Плюс, 2009.
6. [http://www.python-course.eu/python3\\_object\\_oriented\\_programming.php](http://www.python-course.eu/python3_object_oriented_programming.php)
7. <http://foobarnbaz.com/2012/07/08/understanding-python-variables/>

### **ТЕМА № 14. Рекурсивні структури даних.**

**Лекція 19.** Стеки, черги та деки. Списки – 2 год.

Статичні та динамічні структури даних. Рекурсивні структури даних. Стеки деки та черги. Реалізація стеку та черги на базі списку. Реалізація деку на базі посилань на об'єкти. Списки. Реалізація кільцевого списку.

**Лабораторне заняття № 19.** Стеки, черги та деки. Списки – 2 год.

**Лекція 20.** Графи та дерева – 2 год.

Графи та дерева. Реалізація бінарного дерева на базі посилань на об'єкти.  
Реалізація графа на базі словника. Перевизначення операцій

**Лабораторне заняття № 20.** Графи та дерева. – 2 год.

**Самостійна робота з вивчення матеріалу лекцій № 19-20** - 2 год.

#### **Література**

1. Бублик В.В., Личман В.В., Обвінцев О.В.. Інформатика та програмування. Електронний конспект лекцій, 2003 р.,
2. Марк Лутц, Изучаем Python, 4-е издание, 2010, Символ-Плюс
3. Python 3.4.3 documentation
4. Bruno R. Preiss, Data Structures and Algorithms with Object-Oriented Design Patterns in Python, 2003, <http://www.brpreiss.com/books/opus7/>
5. [http://www.python-course.eu/graphs\\_python.php](http://www.python-course.eu/graphs_python.php)
6. [http://www.python-course.eu/python3\\_magic\\_methods.php](http://www.python-course.eu/python3_magic_methods.php)
7. <http://www.programiz.com/python-programming/operator-overloading>

**ТЕМА № 15. Обробка помилок та виключних ситуацій.**

**Лекція 21.** Обробка помилок та виключних ситуацій – 2 год.

Поняття помилки та виключної ситуації. Засоби обробки виключень у Python. Блоки try – except. Атрибути виключень, ініціювання виключень. Стандартні класи виключень у Python. Власні виключення. Твердження про програми

**Лабораторне заняття № 21.** Обробка помилок та виключних ситуацій. – 2 год.

**Самостійна робота з вивчення матеріалу лекції № 21**– 2 год.

#### **Література**

1. A Byte of Python (Russian) Версія 2.01 Swaroop С Н (Translated by Vladimir Smolyar),  
<http://wombat.org.ua/AByteOfPython/AByteofPythonRussian-2.01.pdf>
2. Марк Лутц, Изучаем Python, 4-е издание, 2010, Символ-Плюс
3. Python 3.4.3 documentation
4. Марк Саммерфилд, Программирование на Python 3. Подробное руководство. - Символ-Плюс, 2009.
5. [http://www.python-course.eu/python3\\_exception\\_handling.php](http://www.python-course.eu/python3_exception_handling.php)

## ЗМІСТОВИЙ МОДУЛЬ 4. МЕТАПРОГРАМУВАННЯ.

### ТЕМА № 16. Ітератори та генератори.

**Лекція № 22.** Ітератори та генератори – 2 год.

Ітератори, функції `iter` та `next`. Ітераційний протокол. Генератори. Генератори-вирази та генератори функції. Написання власних класів-ітераторів.

**Лабораторне заняття № 22.** Ітератори та генератори. – 2 год.

**Самостійна робота з вивчення матеріалу лекції № 22**– 2 год.

#### Література

1. Марк Лутц, ИзучаемPython, 4-е издание, 2010, Символ-Плюс
2. Python 3.4.3 documentation
3. Марк Саммерфилд, Программирование на Python 3. Подробное руководство. - Символ-Плюс, 2009.
4. Bruno R. Preiss, Data Structures and Algorithms with Object-Oriented Design Patterns in Python, 2003, <http://www.brpreiss.com/books/opus7/>
5. TarekZiadé. ExpertPythonProgramming. - PacktPublishing, 2008.
6. DavidBeazleyandBrian K. Jones, PythonCookbook. -O'ReillyMedia, 2013.
7. [http://www.python-course.eu/python3\\_generators.php](http://www.python-course.eu/python3_generators.php)

### ТЕМА № 17. Декоратори.

**Лекція 23.** Декоратори – 2 год.

Визначення декоратора. Застосування стандартних декораторів `@staticmethod`, `@property`. Написання власних декораторів. Написання власних декораторів з параметрами. Вкладені декоратори

**Лабораторне заняття № 23.** Декоратори – 2 год.

**Самостійна робота з вивчення матеріалу лекції № 23**– 2 год.

#### Література

1. Марк Лутц, ИзучаемPython, 4-е издание, 2010, Символ-Плюс
2. Python 3.4.3 documentation
3. Марк Саммерфилд, Программирование на Python 3. Подробное руководство. - Символ-Плюс, 2009.
4. TarekZiadé. ExpertPythonProgramming. - PacktPublishing, 2008.
5. DavidBeazleyandBrian K. Jones, PythonCookbook. - O'ReillyMedia, 2013.
6. <https://wiki.python.org/moin/PythonDecoratorLibrary>
7. <http://habrahabr.ru/post/141501/>



8. [http://www.linuxtopia.org/online\\_books/programming\\_books/python\\_programming/python\\_ch26s05.html](http://www.linuxtopia.org/online_books/programming_books/python_programming/python_ch26s05.html)
9. <http://thecodeship.com/patterns/guide-to-python-function-decorators/>
10. <http://www.ibm.com/developerworks/library/l-cpdecor/>

## **ТЕМА № 18. Множинне наслідування.**

**Лекція 24.** Множинне наслідування – 2 год.

Множинне наслідування. Синтаксис множинного наслідування. Менеджер контексту with. Функція super. Доступ до методів класів – предків. Порядок співставлення методів (MRO). Обмеження множинного наслідування та super. Класи-домішки (mixins)

**Лабораторне заняття № 24.** Множинне наслідування – 2 год.

**Самостійна робота з вивчення матеріалу лекції № 24**– 2 год.

### **Література**

1. Марк Лутц, ИзучаемPython, 4-е издание, 2010, Символ-Плюс
2. Python 3.4.3 documentation
3. Марк Саммерфилд, Программирование на Python 3. Подробное руководство. - Символ-Плюс, 2009.
4. Bruno R. Preiss, Data Structures and Algorithms with Object-Oriented Design Patterns in Python, 2003, <http://www.brpreiss.com/books/opus7/>
5. Tarek Ziadé. Expert Python Programming. - Packt Publishing, 2008.
6. David Beazley and Brian K. Jones, PythonCookbook. - O'ReillyMedia, 2013.
7. <http://habrahabr.ru/post/62203/>

## **ТЕМА № 19. Метакласи та метапрограмування.**

**Лекція 25.** Метакласи. Абстрактні класи – 2 год.

Метакласи. Оголошення метакласу у класі. Абстрактні класи

**Лабораторне заняття № 25** Метакласи. Абстрактні класи – 2 год.

**Лекція 26.** Метапрограмування – 2 год.

Метапрограмування. Декоратори класів. Класові методи. Побудова класів у динаміці. Написання власних метакласів

**Лабораторне заняття № 25** Метапрограмування – 2 год.

## Самостійна робота з вивчення матеріалу лекцій №№ 25-26– 2 год.

### Література

1. Марк Лутц, Изучаем Python, 4-е издание, 2010, Символ-Плюс
2. Python 3.4.3 documentation
3. Марк Саммерфилд, Программирование на Python 3. Подробное руководство. - Символ-Плюс, 2009.
4. Tarek Ziadé. Expert Python Programming. - Packt Publishing, 2008.
5. David Beazley and Brian K. Jones, Python Cookbook. - O'Reilly Media, 2013.
6. <http://python-3-patterns-idioms-test.readthedocs.org/en/latest/Metaprogramming.html#basic-metaprogramming>
7. <http://blog.ionelmc.ro/2015/02/09/understanding-python-metaclasses/>
8. <http://habrahabr.ru/post/65625/>
9. [http://www.python-course.eu/python3\\_metaclasses.php](http://www.python-course.eu/python3_metaclasses.php)
10. <http://lgiordani.com/blog/2014/10/14/decorators-and-metaclasses/#.VpFyGfmLTdc>
11. <http://eli.thegreenplace.net/2011/08/14/python-metaclasses-by-example>

## ПИТАННЯ ДО ІСПИТУ

### I Семестр

1. Визначення інформатики та інформації
2. Поняття алгоритму та виконавця
3. Програми та програмування, компілятори та інтерпретатори
4. Основні команди Python
5. Ланцюги та лінійні програми
6. Рівносильність інструкцій
7. Правила запису програм
8. Алгебра висловлювань, бульові операції, їх властивості
9. Відношення та їх властивості
10. Умови
11. Розгалуження (звичайне розгалуження, захищена команда, каскадне розгалуження)
12. Розгалужені програми
13. Поняття циклу та циклічної програми
14. Цикл з умовою продовження, його властивості.
15. Рекурентні співвідношення 1 порядку.
16. Рекурентні співвідношення вищих порядків.
17. Системи рекурентних співвідношень.
18. Повний синтаксис циклу за умовою, обчислення границь
19. Цикл по діапазону значень.
20. Поняття типу даних. Складові частини типу даних.
21. Поziційний запис дійсного числа.
22. Функція округлення, похибка округлення.
23. Наближені операції та їх властивості
24. Цілий тип даних.
25. Дійсний тип даних.
26. Комплексний тип даних
27. Символи. Стандарти кодування символів.
28. Unicode. Кодування символів у Unicode та формати представлення символів.
29. Рядки: носій, операції, відношення та інструкції.
30. Рядки як послідовності
31. Вирізки та їх використання.
32. Додаткові функції для рядків.
33. Списки: носій, операції, відношення та інструкції для списків.
34. Реалізація масивів на базі списків
35. Взаємозв'язок між рядками та списками
36. Спискоутворення. Функція enumerate
37. Кортежі. Носій для кортежів.
38. Операції, відношення та інструкції для кортежів.
39. Пакування та розпакування кортежів
40. Функції zip, sum та map.
41. Іменовані кортежі.

- 42.Словники. Носій для словників.
- 43.Операції, відношення та інструкції для словників.
- 44.Функції all any та filter
- 45.Словникоутворення.
- 46.Означення підпрограми.
- 47.Простий синтаксис та просте правило виклику функції
- 48.Передача параметрів різних типів у підпрограми
- 49.Функція format
- 50.Оформлення документації функції
- 51.Повернення функцією декількох результатів
- 52.Значення параметрів за угодою
- 53.Позиційні та ключові параметри
- 54.Змінна кількість параметрів функції
- 55.Повний синтаксис та повне правило виклику функції
- 56.lambda-функції
- 57.Рекурсія та рекурсивні підпрограми
- 58.Глобальні змінні
- 59.Функціональний тип даних
- 60.Означення модуля.
- 61.Модулі у Python
- 62.Імпорт модулів та використання об'єктів модулів
- 63.Видимість об'єктів модуля
- 64.Області дії імен
- 65.Пакети
- 66.Рядки документації модуля
- 67.Множини: носій, основні операції, відношення та інструкції для множин.
- 68.Додаткові операції, відношення та інструкції для множин.
- 69.Множини, що не змінюються.
- 70.Множиноутворення.

## ПИТАННЯ ДО ЗАЛІКУ

### II Семестр

1. Файли, носій типу файл, текстові та нетекстові файли.
2. Відкриття та закриття файлів.
3. Читання з файлу та запис у файл.
4. Додаткові дії для файлів.
5. Запис та читання з текстового файлу нетекстових значень.
6. Параметри при запуску програми з командного рядка.
7. Обробка нетекстових файлів: pickle та shelve.
8. Об'єктно-орієнтований підхід до програмування.
9. Визначення класів та об'єктів, інші поняття ООП
10. Синтаксис опису класів у Python
11. Створення та використання об'єктів класу.
12. Зв'язування між об'єктами та методами. Віртуальні методи.
13. Обмеження доступу до полів та методів класу
14. Рядки документації класу
15. Статичні поля та статичні методи класів
16. Статичні та динамічні структури даних. Рекурсивні структури даних
17. Стеки деки та черги.
18. Реалізація стеку та черги на базі списку.
19. Реалізація деку на базі посилань на об'єкти
20. Списки. Реалізація кільцевого списку.
21. Графи та дерева
22. Реалізація бінарного дерева на базі посилань на об'єкти
23. Реалізація графа на базі словника
24. Перевизначення операцій
25. Поняття помилки та виключної ситуації.
26. Засоби обробки виключень у Python
27. Блоки try – except
28. Атрибути виключень, ініціювання виключень.
29. Стандартні класи виключень у Python.
30. Власні виключення
31. Твердження про програми
32. Ітератори, функції `iter` та `next`
33. Ітераційний протокол
34. Генератори
35. Генератори-вирази та генератори функції
36. Написання власних класів-ітераторів
37. Визначення декоратора
38. Застосування стандартних декораторів `@staticmethod`, `@property`
39. Написання власних декораторів
40. Написання власних декораторів з параметрами
41. Вкладені декоратори
42. Множинне наслідування. Синтаксис множинного наслідування
43. Менеджер контексту `with`

44. Функція `super`
45. Доступ до методів класів –предків.
46. Порядок співставлення методів (MRO)
47. Обмеження множинного наслідування та `super`
48. Класи-домішки (mixins)
49. Метакласи. Оголошення метакласу у класі.
50. Абстрактні класи
51. Метапрограмування
52. Декоратори класів
53. Класові методи
54. Побудова класів у динаміці
55. Написання власних метакласів

## ДЖЕРЕЛА ІНФОРМАЦІЇ

### Книги до курсу

1. A Byte of Python (Russian) Версия 2.01 Swaroop С Н (Translated by Vladimir Smolyar),  
<http://wombat.org.ua/AByteOfPython/AByteofPythonRussian-2.01.pdf>
2. Марк Лутц, ИзучаемPython, 4-е издание, 2010, Символ-Плюс
3. Python 3.4.3 documentation. <https://docs.python.org/3.4/index.html>
4. Бублик В.В., Личман В.В., Обвінцев О.В.. Інформатика та програмування. Електронний конспект лекцій, 2003 р.,  
<http://www.matfiz.univ.kiev.ua/books> (також на  
<http://obvintsev.info/compuscience/lectures/index.htm>)
5. Самоучитель Python. <http://pythonworld.ru/samouchitel-python>
6. С. Шапошникова. Основы программирования на Python. Версия 2 (2011). <http://younglinux.info/pdf>
7. Бизли Д. Python. Подробный справочник. - Символ-Плюс, 2010.
8. Марк Саммерфилд, Программирование на Python 3. Подробное руководство. - Символ-Плюс, 2009.
9. Doug Hellmann. Python Module of the Week. - <http://pymotw.com/2/#>
10. Matt Telles. Python Power! The Comprehensive Guide. - Thomson Course Technology, 2012.
11. Mark Lutz. Programming Python 4th Edition. - O'Reilly Media, 2010.
12. Tarek Ziadé. Expert Python Programming. - Packt Publishing, 2008.
13. Bernd Klein. Python Course <http://www.python-course.eu/index.php>

### Книги з програмування

1. Вирт Н. Алгоритмы + структуры данных = программы.-М.: Мир, 1985.- 406с.
2. Грис Д. Наука программирования .-М.: Мир, 1984.- 416с.
3. Кнут Д.Э. Искусство программирования (в 3 томах). – М.: Издательский дом «Вильямс», 2002

### Україномовні ресурси з Python

1. Марк Пілігрим. Пориньте у Python 3.  
[https://uk.wikibooks.org/wiki/%D0%9F%D0%BE%D1%80%D0%B8%D0%BD%D1%8C%D1%82%D0%B5\\_%D1%83\\_Python\\_3](https://uk.wikibooks.org/wiki/%D0%9F%D0%BE%D1%80%D0%B8%D0%BD%D1%8C%D1%82%D0%B5_%D1%83_Python_3)

2. Програмування на мовіPython (3.x). Початковий курс .  
<https://sites.google.com/site/pythonukr/>
3. Віталій Подоба. Python 2: Курс Молодого Бійця.  
<http://www.vitaliypodoba.com/tutorials/python2-beginners-course/>
4. Віталій Подоба. Веб-розробка з Python та Django для початківців.  
<http://www.vitaliypodoba.com/books/django-for-beginners/>

Затверджено на засіданні кафедри математичної фізики, протокол № \_\_\_\_ від \_\_\_\_\_.\_\_\_\_.2015 р.

Завідувач кафедри  
професор

В.Г. Самойленко