



# РОЗРОБКА СИСТЕМНИХ ПРОГРАМ

## Робоча програма навчальної дисципліни (Силабус)

### Реквізити навчальної дисципліни

Рівень вищої освіти	<i>Перший (бакалаврський)</i>
Галузь знань	<i>12 Інформаційні технології</i>
Спеціальність	<i>123 Комп'ютерна інженерія</i>
Освітня програма	<i>Комп'ютерні системи та мережі</i>
Статус дисципліни	<i>Вибіркова</i>
Форма навчання	<i>очна(денна)</i>
Рік підготовки, семестр	<i>3 курс, осінній семестр</i>
Обсяг дисципліни	<i>4 кредити (ECTS). Обсяг дисципліни - 120 годин, у тому числі 54 аудиторних години та 66 годин самостійної роботи студентів</i>
Семестровий контроль/ контрольні заходи	<i>5 семестр – залік, МКР</i>
Розклад занять	
Мова викладання	<i>Українська / Англійська</i>
Інформація про керівника курсу / викладачів	<i>к.т.н., доцент Павлов Валерій Георгійович, pavlovvg@ukr.net</i>
Розміщення курсу	<i><a href="https://campus.kpi.ua/tutor/index.php?mode=mob&amp;show&amp;irid=218710">https://campus.kpi.ua/tutor/index.php?mode=mob&amp;show&amp;irid=218710</a></i>

### Програма навчальної дисципліни

#### 1. Опис навчальної дисципліни, її мета, предмет вивчення та результати навчання

Навчальна дисципліна "**Розробка системних програм**" відноситься до вибіркового освітнього компоненту навчальної програми, а саме до циклу "Вибіркові освітні компоненти з міжфакультетського/факультетського/кафедрального Ф-каталогів". Вона має номер **12** у переліку компонентів освітньої програми.

**Причини та мотивація до вивчення:** необхідність навчальної дисципліни "Розробка системних програм" є закріплення, поглиблення, узагальнення теоретичних знань та практичних навичок, які отримують студенти під час вивчення дисципліни "Системне програмування", розуміння принципів програмного керування безпосередньо процесору.

**Мета навчальної дисципліни:** Одержання навичок складання системних програм у відповідності до вимог завдання або технічного завдання і підготовки комплекту документів на програми для її замовника або користувача. Вивчення основних вимог до базових документів системних програм та одержання навичок оформлення таких документів на прикладах реалізації системних програм.

**Метою навчальної дисципліни** є формування у студентів низки компетентностей, а саме:

#### **ЗДАТНІСТЬ:**

- аналізувати процеси, які здійснюються під час трансляції та компіляції;
- розробляти системне програмне забезпечення за допомогою сучасних інструментальних засобів розробки програмного забезпечення;

- налагоджувати системні програми.

## 1.2. Основні завдання навчальної дисципліни.

Після засвоєння навчальної дисципліни студенти мають продемонструвати такі результати навчання:

### **ЗНАННЯ:**

- взаємодії системних програм під час їх виконання;
- методів опису мов спілкування та мов програмування зокрема;
- моделей та реалізацій контекстно-незалежних та автоматних граматики;
- особливостей та етапів розробки системних програм;
- методів рішення задач лексичної, синтаксичної та семантичної обробки вхідних кодів програм.

### **УМІННЯ:**

- використовувати потрібні інструментальні засоби для вирішення завдань;
- розробити або виконати настройку алгоритму синтаксичного аналізу,
- описати внутрішню форму представлення програм в вигляді графів;
- здійснювати опис та виконувати обробку конкретних операторів комп'ютерних мов.

### **НАВИЧКИ:**

- проведення експериментів, збирання даних та моделювання в комп'ютерних системах;
- спілкуватись усно та письмово з професійних питань українською мовою та однією з іноземних мов (англійською, німецькою, італійською, французькою, іспанською);
- адаптуватись до нових ситуацій, обґрунтовувати, приймати та реалізовувати у межах компетенції рішення.

## **2. Пререквізити та постреквізити дисципліни (місце в структурно-логічній схемі навчання за відповідною освітньою програмою)**

Міждисциплінарні зв'язки: Для успішного вивчення дисципліни «Розробка системних програм» студенти повинні засвоїти матеріал та мати певні знання, вміння та навички з таких дисциплін, як:

- ЗО9 - «Програмування»,
- ПО1 - «Комп'ютерна логіка»,
- ПО6 - «Архітектура комп'ютерів»,
- ПО13 - «Алгоритми та методи обчислень»,
- ПО 7- «Системне програмування».

Знання та навички, які отримуються під час вивчення дисципліни «Курсова робота з Системного програмування», можуть бути використані в подальшому при опануванні наступних курсів:

- ПО6 – «Архітектура комп'ютерів»,
- ПО9 - «Системне програмне забезпечення»,
- ПО17 – «Курсова робота з Архітектури комп'ютерів»,
- ПО18 – «Курсова робота з Системне програмне забезпечення», а також під час дипломного проектування.

## **3. Зміст навчальної дисципліни**

Розділ 1. Методи обробки і перетворень в програмах трансляції

### *Тема 1.1 Структура системного програмного забезпечення*

- *Предмет та задачі курсу.*
- *Склад та призначення системного програмного забезпечення (СПЗ).*
- *Місце та роль трансляторів у СПЗ.*

### *Тема 1.2. Принципи побудови та види компіляторів.*

- *Види трансляторів.*
- *Компілятори та інтерпретатори.*
- *Класифікація компіляторів.*
- *Основні складові компіляторів та послідовність компіляції.*
- *Принципи побудови лексичних таблиць та дерев граматичного розбору.*

### *Тема 1.3. Автоматизація створення компіляторів*

- *Опис компіляторів за допомогою T-діаграм.*
- *Зовнішній (front-end) та внутрішній (back-end) інтерфейс.*
- *Метод розкритки, крос-компіляція, віртуальні машини, JIT-компілятори.*
- *Система YACC.*

## *Розділ 2. Методи формалізації опису мов програмування*

### *Тема 2.1. Методи опису природних та штучних мов. Метамови.*

- *Поняття мови.*
- *Засоби опису мов.*
- *Природні та штучні мови.*
- *Поняття абетки та граматики мови.*
- *Метамови БНФ та РБНФ.*

### *Тема 2.2. Породжуючи граматики.*

- *Поняття породжуючих граматик за Н. Хомським.*
- *Сентенціальна форма виводу.*
- *Основні складові опису граматик.*
- *Приклади опису різних мов за допомогою породжуючи граматик*

### *Тема 2.3. Класифікація формальних граматик. Побудова синтаксичних дерев.*

- *Класифікація граматик за Н. Хомським.*
- *Типи розпізнавачів.*
- *Правила побудови синтаксичних дерев.*

## *Розділ 3. Перетворення контекстно-вільних граматик.*

### *Тема 3.1. Методи перетворення граматик.*

- *Приведена форма контекстно-вільних (КВ) граматик .*
- *Алгоритми видалення  $\epsilon$ -правил та ланцюгових правил.*
- *Алгоритми видалення непороджуючих та недосяжних символів.*
- *Обов'язкова послідовність отримання приведеної форми КВ граматик.*
- *Перетворення КВ граматик у нормальну форму Хомського.*

- Нормальна форма Грейбах.
- Неоднозначність KB – граматик та поняття направляючих символів.
- S-граматика.
- Q-граматика.
- LL(1)-граматика.
- Перетворення KB - граматик до типу LL(1).
- Алгоритм усунення лівої рекурсії.
- Ліва факторизація.

#### Розділ 4. Класи синтаксичних аналізаторів.

##### Тема 4.1. Розпізнавач для регулярних граматик.

- Побудова розпізнавача на основі кінцевого автомату (КА).
- Види опису КА.
- Детерміновані (ДКА) та не детерміновані (НКА) кінцеві автомати
- Алгоритм перетворення НКА в ДКА.
- Регулярні вирази та регулярні множини.
- Алгоритм мінімізації кінцевих автоматів.

##### Тема 4.2. Розпізнавач для KB - граматик.

- Визначення автомату з магазинною пам'яттю (МП - автомата).
- Детерміновані та недетерміновані МП - автомати.
- Типи граматик передування.

## 4. Навчальні матеріали та ресурси

### 4.1. Базова:

1. Abdulaziz Ghuloum. *An Incremental Approach to Compiler Construction, Conference: Scheme and Functional Programming Workshop*, 2006. URL: <http://scheme2006.cs.uchicago.edu/11-ghuloum.pdf>.
2. *Проектування та моделювання програмного забезпечення сучасних інформаційних систем* / Г. В. Табунщик, Т.І. Каплієнко, О.А. Петрова – Запоріжжя : Дике Поле, 2016. – 250 с.,

### 4.2 Допоміжна:

1. Morgan Kaufmann. *Engineering a Compiler, Second Edition (English) (вид. 2 edition)*. 2011-02-21. ISBN 9780120884780.
2. Morgan Kaufmann. *Advanced Compiler Design and Implementation (English) (вид. 1 edition)*. 1997-08-15. ISBN 9781558603202.
3. *Compiler Design in C (English)*. Prentice-Hall. 1990-01-01. ISBN 9780131550452.
4. *Modern Compiler Implementation in Java (English) (вид. 2nd edition)*. Cambridge University Press. 2002-10-21. ISBN 9780521820608.

## Навчальний контент

### 5. Методика опанування навчальної дисципліни (освітнього компонента)

Вивчення дисципліни складається з чотирьох етапів, які наведені в таблиці 5.1.

Таблиця 5.1

Назви розділів і тем	Кількість годин	
	Всього	у тому числі

		Лекції	Практичні (семінарські)	Лабораторні (комп'ютерний і практикум)	СРС
<b>Розділ 1. Методи обробки і перетворень в програмах трансляції</b>					
Тема 1.1. Структура системного програмного забезпечення	4	2	–	–	2
Тема 1.2. Принципи побудови та види компіляторів	8	2	–	2	4
Тема 1.3. Автоматизація створення компіляторів	4	2	–	–	2
<b>Розділ 2. Методи формалізації опису мов програмування</b>					
Тема 2.1. Методи опису природних та штучних мов. Метамови	8	2	–	2	4
Тема 2.2. Породжуючі граматики	8	4	–	–	4
Тема 2.3. Класифікація формальних граматики. Побудова синтаксичних дерев	8	2	–	2	4
Контрольна робота 1	2	–	–	–	2
Разом за розділами 1-2	42	14	–	8	22
<b>Розділ 3. Перетворення контекстно-вільних граматики.</b>					
Тема 3.1. Методи перетворення граматики.	28	10	–	4	14
<b>Розділ 4. Класи синтаксичних аналізаторів.</b>					
Тема 4.1. Розпізнавач для регулярних граматики.	20	6	–	4	10
Тема 4.2. Розпізнавач для КВ - граматики.	20	6	–	4	10
Контрольна робота 2	2	–	–	–	2
Разом за розділами 3-4	70	22	–	12	36
Підготовка до заліку	8	–	–	–	8
<b>Всього годин</b>	<b>120</b>	<b>36</b>	<b>–</b>	<b>18</b>	<b>66</b>

### **Лабораторні заняття**

Основні завдання циклу лабораторних занять є надбання студентами необхідних практичних навичок розробки та аналізу алгоритмів системних програм, розробки та налагодження системних програм з застосуванням мов високого рівня і Асемблера.

№ з/п	Назва лабораторної роботи (комп'ютерного практикуму)	Кількість ауд. годин
1	Розробка та опрацювання базової структури компілятора.	2
2	Унарні та бінарні операції.	2
3	Обробка змінних.	2
4	Вкладені конструкції.	4
5	Тернарний оператор, оператори розгалуження.	4
6	Цикли.	4
	<b>Разом:</b>	<b>18</b>

## 6. Самостійна робота

№ з/п	Назва теми, що виноситься на самостійне опрацювання	Кількість годин СРС
1	Виконання завдань за темою кожного лекційного заняття (див. розділ 4) – 1 год. на 1 годину лекції	36
2	Підготовка до лабораторних занять (див. розділ 5) – 1 година на 1 годину лабораторного заняття	18
3	Підготовка до контрольних (див. розділ 9) – 2 години на кожну МКР.	4
4	Підготовка до заліку	8
	<b>Разом:</b>	66

## Політика та контроль

### 7. Політика навчальної дисципліни (освітнього компонента)

При зарахування та оцінювання лабораторних робіт беруться до уваги наступні чинники:

- Повнота виконання завдання на лабораторну роботу за індивідуальним варіантом;
- Своєчасність виконання лабораторної роботи згідно графіку;
- Самостійність виконання лабораторної роботи та відсутність ознак плагіату;
- Відповіді на питання щодо змісту лабораторної роботи під час її захисту.

При оцінюванні контрольних робіт до уваги приймаються:

- Правильність та повнота виконання завдань;
- Кількість виконаних завдань в умовах обмеженого часу;
- Самостійність виконання завдань та відсутність ознак плагіату;
- Кількість спроб виконання контрольних, які передують тій, що оцінюється.

Для підготовки до контрольних студенти отримують перелік теоретичних питань та зміст типових задач, які будуть у завданнях на контрольних.

При першій та другій атестації до уваги приймається кількість лабораторних робіт та контрольних робіт зарахованих на час проведення атестації.

### 8. Види контролю та рейтингова система оцінювання результатів навчання (PCO)

Система оцінювання успішності студентів з дисципліни «Розробка системних програм» ґрунтується на «Положенні про систему оцінювання результатів навчання в КПІ ім. Ігоря Сікорського» ([https://document.kpi.ua/files/2020\\_1-273.pdf](https://document.kpi.ua/files/2020_1-273.pdf)), а саме Рейтингової системі оцінювання (PCO) першого типу (PCO-1).

PCO з дисципліни, семестровий контроль з якої передбачений у вигляді заліку, розробляється за типом PCO-1 і включає оцінювання заходів поточного контролю з дисципліни впродовж семестру.

Рейтингова оцінка здобувача складається з балів, отриманих здобувачем за результатами заходів поточного контролю (лабораторних робіт ( $R_L$ ) і контрольних ( $R_K$ )), заохочувальних ( $R_3$ ) та штрафних ( $R_{\text{ш}}$ ) балів:

$$R_S = R_L + R_K + R_3 + R_{\text{ш}}$$

де  $R_L$  за 6 лабораторних робіт складає  $6 \times 10 = 60$  балів,

$R_K$  за дві контрольні складає  $2 \times 20 = 40$  балів.

Таким чином максимальна основна сума балів за семестр складає  $60 + 40 = 100$  балів.

*Поза основною шкалою оцінювання передбачені заохочувальні та штрафні бали, які враховуються у загальній сумі балів, але не входять до основної шкали РСО.*

*Заохочувальні бали враховують відповідь на питання та виконання завдань на лекційних заняттях, якість конспекту.*

*Штрафні бали передбачені за несвоєчасне виконання лабораторних робіт, тобто зі запізненням відносно графіку.*

*Оцінювання результатів навчання здійснюється за 100-бальною шкалою з подальшим переведенням оцінок за університетською шкалою згідно таблиці:*

<i>Кількість балів</i>	<i>Оцінка</i>
100-95	Відмінно
94-85	Дуже добре
84-75	Добре
74-65	Задовільно
64-60	Достатньо
Менше 60	Незадовільно
Не виконані умови допуску	Не допущено

**Робочу програму навчальної дисципліни (силабус):**

**Складено** к.т.н., доцент, Павлов В.Г.

**Ухвалено кафедрою обчислювальної техніки** (протокол № 10 від 25 травня 2022)

**Погоджено** Методичною комісією факультету ФІОТ (протокол № 10 від 09 червня 2022)