Національний технічний університет України
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені ІГОРЯ СІКОРСЬКОГО»

Кафедра інженерії
програмного забезпечення в
енергетиці

ОСНОВИ РОЗРОБКИ ТРАНСЛЯТОРІВ

Робоча програма навчальної дисципліни (Силабус)

Реквізити навчальної дисципліни

Рівень вищої освіти	**Перший (бакалаврський)**
Галузь знань	12 Інформаційні технології
Спеціальність	121 Інженерія програмного забезпечення
Освітня програма	Інженерія програмного забезпечення інтелектуальних кібер-фізичних систем в енергетиці
Статус дисципліни	Нормативна
Форма навчання	очна (денна)
Рік підготовки, семестр	3 курс осінній семестр
Обсяг дисципліни	7,5 кредитів ECTS /225 годин (54 години лекцій, 54 години практичних занять, 117 годин — самостійна робота студента)
Семестровий контроль/ контрольні заходи	Екзамен, МКР, РГР, представлення робіт комп'ютерного практикуму
Розклад занять	http://rozklad.kpi.ua
Мова викладання	Українська
Інформація про керівника курсу / викладачів	Лектор: к.т.н., доцент, Стативка Юрій Іванович, statyvka-yu@lll.kpi.ua Практичні: к.т.н., доцент, Стативка Юрій Іванович, statyvka-yu@lll.kpi.ua
Розміщення курсу	https://ecampus.kpi.ua

Програма навчальної дисципліни

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

В курсі розглядаються як фундаментальні принципи створення і імплементації мов програмування загального призначення, так і практичні аспекти автоматизації розробки трансляторів.

Значна увага приділяється проєктуванню мов програмування, їх специфікації, математичним основам представлення та опрацювання формальних мов.

Розглядаються загальні та спеціалізовані засоби побудови компіляторів та інтерпретаторів: алгоритми лексичного, синтаксичного та семантичного аналізу, генерування коду, засобів виконання коду цільової мови.

Метою дисципліни є опанування студентами теоретичних знань та набуття практичного досвіду проєктування і імплементації імперативних мов програмування .

Предмет дисципліни — методи та алгоритми побудови, специфікації та імплементації мов програмування високого рівня.

Завдання. В результаті вивчення дисципліни у студентів повинні сформуватися наступні компетентності:

фахові:

- здатність брати участь у проектуванні програмного забезпечення, включаючи проведення моделювання (формальний опис) його структури, поведінки та процесів функціонування (ФК 2);
- Здатність розробляти архітектури, модулі та компоненти програмних систем (ФК 3)
- здатність застосовувати транслятори мов програмування при реалізації програмних систем (ФК 19).

Після засвоєння навчальної дисципліни студенти мають продемонструвати такі програмні результати навчання:

- застосовувати на практиці фундаментальні концепції, парадигми і основні принципи функціонування мовних, інструментальних і обчислювальних засобів інженерії програмного забезпечення (ПРН 32).

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

Пререквізити дисципліни. Знання, отримані при вивченні дисциплін: «Компоненти програмної інженерії», «Основи програмування», «Комп'ютерна дискретна математика», «Алгоритми та структури даних».

Постреквізити дисципліни. Компетенції, отримані студентами в процесі вивчення дисципліни можуть бути використані при виконанні дипломної роботи.

3. Зміст навчальної дисципліни

Розділ 1. Основи теорії формальних мов

Тема 1.1. Вступ. Основні блоки та структура компілятора.
Тема 1.2. Елементи теорії формальних граматик.
Тема 1.3. Приведення граматик.
Тема 1.4.Форми подання граматики.
Тема 1.5. Специфікація формальних мов.
Тема 1.6. Регулярні вирази та мови.
Тема 1.7. Скінченні автомати.

Розділ 2. Лексичний, синтаксичний та семантичний аналіз програм

Тема 2.1. Лексичний аналіз програм.
Тема 2.2. Приклад реалізації лексичного аналізатора на основі діаграми станів.
Тема 2.3. Автомат з магазинною пам'яттю (МП-автомат).
Тема 2.4.Призначення та види синтаксичного аналізу. Рекурсивний спуск
Тема 2.5. Реалізація синтаксичного аналізатора на основі алгоритму рекурсивного спуску.
Тема 2.6. Використання автомата з магазинною пам'яттю для синтаксичного розбору.
Тема 2.7. Семантичний аналіз програм.
Тема 2.8 Приклад реалізації семантичного аналізатора.

Розділ 3. Трансляція та виконання програм

Тема 3.1. Проміжні форми подання програм.
Тема 3. 2. Трансляція у ПОЛІЗ.
Тема 3.3. Виконання ПОЛІЗ-програм: Postfix State Machine (PSM).
Тема 3.4. Common Languge Runtime (CLR).
Тема 3.5. Переклад на Common Intermediate Languge (CIL).
Тема 3.6. Java Virtual Machine (JVM).
Тема 3.7. Переклад на Java-асемблер.

Розділ 4. Автоматизація розробки трансляторів

Тема 4.1. Засоби автоматизації розробки трансляторів
Тема 4.2. Синтаксичне дерево в ANTLR4.
Тема 4.3. ANTLR4: target language Java
Тема 4.4. ANTLR4: target language Python
Тема 4.5. Розробка трансляторів: підсумки.

4. Навчальні матеріали та ресурси

Основна література

1. Стативка Ю.І. Формальні мови: Основні концепти і представлення [Текст]: навч. посіб. / Ю. І. Стативка. – Київ: КПІ ім. Ігоря Сікорського, 2023. – 87 с.
2. Медведєва В.М. Транслятори: лексичний та синтаксичний аналізатори [Текст] : навч. посіб. / В.М. Медведєва, В.А. Третяк. – К. : НТУУ «КПІ», 2012. – 148с.
3. Медведєва В.М. Транслятори: внутрішнє подання програм та інтерпретація [Текст] : навч. посіб. / В.М. Медведєва, В.А. Третяк. . – К. : Текст, 2015. – 144 с.
4. Aho, Alfred, Lam, Monica, Sethi, Ravi, Ullman, Jeffrey Compilers: Principles, Techniques, and Tools, 2nd edition. - Addison Wesley, 2006. - 1040 p.

Додаткова література

5. Hopcroft J. E., Motwani R., Ullman J. D. Introduction to Automata Theory, Languages, and Computation (3rd ed.). Pearson, 2013. 560 p.
6. A. V. Aho and J. D. Ullman, The Theory of Parsing, Translation, and Compiling, Vol. 1, Parsing. Prentice Hall, 1972. 1030 p.
7. Winskel G. The formal semantics of programming languages: an introduction. Cambridge, Massachusetts, London: MIT Press, 1993. 384 p.

Інформаційні ресурси

8. https://www.antlr.org/
9. https://github.com/antlr/antlr4/blob/master/doc/getting-started.md
10. https://github.com/antlr/antlr4/blob/4.6/doc/index.md
11. EBNF Visualizer: веб-сайт. URL: http://dotnet.jku.at/applications/visualizer/
12. EBNF 2 RailRoad: веб-сайт. URL: https://github.com/matthijsgroen/ebnf2railroad
13. Railroad Diagram Generator: веб-сайт. URL: https://bottlecaps.de/rr/ui
14. JFLAP Version 7.1 RELEASED July 27, 2018: веб-сайт. URL: https://www.jflap.org/ .

15. Rodger S. H., Finley T. W. JFLAP – An Interactive Formal Languages and Automata Package Jones & Bartlett Publishers, Sudbury, 2005. 212 p. – URL: https://www2.cs.duke.edu/csed/jflap/jflapbook/jflapbook2006.pdf.
16. https://learn.microsoft.com/en-us/dotnet/framework/tools/ilasm-exe-il-assembler
17. https://learn.microsoft.com/en-us/dotnet/framework/tools/ildasm-exe-il-disassembler
18. https://github.com/drstrng/Krakatau-noff

Навчальний контент

5. Методика опанування навчальної дисципліни (освітнього компонента)

Лекційні заняття

Розділ 1. Основи теорії формальних мов

Лекція 1. Вступ. Основні блоки та структура компілятора
Місце та призначення компілятора в програмному забезпеченні комп'ютера. Основні блоки та їх взаємодія. Структура компілятора.

Лекція 2. Елементи теорії формальних граматик.
Ланцюжки, операції над ними. Мова. Генеративна граматика. Виведення. Сентенційна форма. Рекурсивні правила.

Лекція 3. Приведення граматик.
Трансформація граматик. Усунення ε-продукцій. Усунення ланцюгових правил. Усунення марних символів. Факторизація граматик. Нормальна форма Чомскі.

Лекція 4.Форми подання граматики.
Нотація Бекуса-Наура. Розширена нотація Бекуса-Наура. Синтаксичне дерево. Абстрактне синтаксичне дерево. Синтаксичні діаграми. Однозначність граматики. Варіанти граматик арифметичного виразу.

Лекція 5. Специфікація формальних мов.
Мови програмування і їх представлення. Структура специфікації — лексика, синтаксис, семантика. Місце семантики у специфікації. Приклади специфікації. Синтаксичні діаграми для представлення синтаксису.

Лекція 6. Регулярні вирази та мови.
Визначення. Метамова представлення регулярних виразів. Граф регулярного виразу. Еквівалентність множини регулярних і автоматних мов.

Лекція 7. Скінченні автомати.
Поняття розпізнавача. Види розпізнавачів. Скінченні автомати. Визначення, правила функціювання, форми подання. Детерміновані скінченні автомати (ДСА). Недетерміновані скінченні автомати (НСА). Мінімізація ДСА. Автоматні мови. Еквівалентність множини регулярних і автоматних мов.

Розділ 2. Лексичний, синтаксичний та семантичний аналіз програм

Лекція 8. Лексичний аналіз програм.

Лексичний аналізатор. Призначення, вхід, вихід. Діаграма станів. Лексичний аналіз програм на основі діаграми станів.

Лекція 9. Приклад реалізації лексичного аналізатора на основі діаграми станів.
Постановка завдання. Приклад виконання.

Лекція 10. Автомат з магазинною пам'яттю (МП-автомат).
Визначення, правила функціювання, форми подання. МП-автомат і контекстно вільні граматики.

Лекція 11.Призначення та види синтаксичного аналізу. Рекурсивний спуск
Призначення, види синтаксичного розбору. Низхідний розбір. Алгоритм рекурсивного спуску. Перетворення граматики для реалізації розбору без повернень та зациклювання.

Лекція 12. Реалізація синтаксичного аналізатора на основі алгоритму рекурсивного спуску.
Приклад реалізації парсера на основі алгоритму рекурсивного спуску.

Лекція 13. Використання автомата з магазинною пам'яттю для синтаксичного розбору.
Нерекурсивний синтаксичний аналіз. Низхідний та висхідний синтаксичний аналіз.

Лекція 14. Семантичний аналіз програм.
Семантика мови і поведінка програми. Аналіз відповідності програми семантиці, декларованій у специфікації мови.

Лекція 15 Приклад реалізації семантичного аналізатора.
Змінні: декларація, ініціація, l- і r-value. Типи змінних, констант, виразів, інструкцій (statements).

Розділ 3. Трансляція та виконання програм

Лекція 16. Проміжні форми подання програм.
Призначення проміжних форм, їх види. Польський інверсний запис. Алгоритм Дейкстри побудови ПОЛІЗ. Пріоритети.

Лекція 17. Трансляція у ПОЛІЗ.
Переклад у ПОЛІЗ конструкцій мови високого рівня.

Лекція 18. Виконання ПОЛІЗ-програм: Postfix State Machine (PSM).
Виконання ПОЛІЗ-програм: стекова машина. Розширення PSM.

Лекція 19. Common Languge Runtime (CLR).
Віртуальна стекова машина .Net. ILASM. ILDASM. Набір інструкцій CLR.

Лекція 20. Переклад на Common Intermediate Languge (CIL).
Побудова транслятора на CIL. Асемблювання і виконання il-програм.

Лекція 21. Java Virtual Machine (JVM).
Віртуальна стекова машина Java. Набір інструкцій JVM. Засоби асемблювання та дизасемблювання для JVM. Асемблери та дизасемблери третіх сторін: Krakatau.

Лекція 22. Переклад на Java-асемблер.
Побудова транслятора для JVM. Асемблювання і виконання j-програм.

Розділ 4. Автоматизація розробки трансляторів

Лекція 23. Засоби автоматизації розробки трансляторів
Огляд генератора парсерів: Yacc, Bison, Lex, Flex, Happy, BNFC, ANTLR4.
ANTLR4: Структура, встановлення, використання. Засоби проєктування та часу виконання. Особливості представлення граматики.

Лекція 24. Синтаксичне дерево в ANTLR4.
Побудова та обхід синтаксичного дерева. Listeners та Visitors.

Лекція 25. ANTLR4: target language Java
Інтегрування згенерованих компонент у Java-застосунок.

Лекція 26. ANTLR4: target language Python
Інтегрування згенерованих компонент у Python-застосунок.

Лекція 27. Розробка трансляторів
Огляд курсу: концепти, технології, інструменти. Набутий досвід у контексті розробки і реалізації мов програмування.

Практичні заняття

Практичне заняття 1. Мови програмування
Мови програмування. Типологія мов програмування. Парадигми. Інструменти розробника. Компілятори та інтерпретатори.

Практичне заняття 2. Основні блоки та структура транслятора.
Основні блоки та структура транслятора. Компілятори і інтерпретатори. Однопрохідний, багатопрохідний компілятор. Фази компіляції.

Практичне заняття 3. Елементи теорії формальних мов.
Формальні мови. Ланцюжки, операції над ними. Формальні граматики. Виведення. Побудова граматики за описом мови. Побудова опису мови за граматикою.

Практичне заняття 4. Приведення граматик.
Усунення ε-продукцій. Усунення ланцюгових правил. Усунення марних символів. Факторизація граматик. Нормальна форма Чомскі.

Практичне заняття 5. Форми подання граматик.
Нотація і розширена нотація Бекуса-Наура. Побудова граматик для елементів мови програмування у нотації РБНФ. Генератори синтаксичних діаграм. EBNF-Vizualizer.

Практичне заняття 6. Специфікація формальних мов.
Структура специфікації — лексика, синтаксис, семантика. Ревю специфікацій.

Практичне заняття 7. Представлення роботи комп’ютерного практикуму № 1 (**КП 1**).

Представлення специфікацій розробленої мови.
Ревю специфікацій розробленої мови програмування.

Практичне заняття 8. Регулярні мови.
Регулярні вирази і мови.
Ревю специфікацій розробленої мови програмування.

Практичне заняття 9. Скінченні автомати.
Детерміновані скінченні автомати (ДСА). Мінімізація ДСА. Недетерміновані скінченні автомати (НСА). Приведення НСА до ДСА. JFLAP. Еквівалентність множини регулярних і автоматних мов.

Практичне заняття 10. Лексичний аналізатор розробленої мови.
Лексичний аналізатор (сканер).
Ревю проєкту сканера розробленої мови програмування. Діаграма станів. Представлення даних у коді.

Практичне заняття 11. Представлення **КП 2**.
Представлення сканера розробленої мови програмування.
Ревю сканера розробленої мови програмування.

Практичне заняття 12. Автомат з магазинною пам'яттю (МП-автомат).
Детерміновані і недетерміновані МП-автомати. JFLAP. МП-автомат і контекстно вільні граматики.

Практичне заняття 13. Предиктивний синтаксичний аналіз методом рекурсивного спуску.
Трансформація граматики для реалізації парсера.
Ревю сканера розробленої мови програмування.

Практичне заняття 14. Предиктивний синтаксичний аналіз методом рекурсивного спуску.
Ревю сканера розробленої мови програмування.

Практичне заняття 15. Семантичний аналізатор для розробленої мови програмування.
Ревю семантичного аналізатора для розробленої мови програмування.

Практичне заняття 16. Представлення **КП 3**.
Представлення синтаксичного і семантичного аналізаторів для розробленої мови програмування.
Ревю синтаксичного і семантичного аналізаторів для розробленої мови програмування.

Практичне заняття 17. Представлення **КП 3**.
Представлення синтаксичного і семантичного аналізаторів для розробленої мови програмування.
Ревю синтаксичного і семантичного аналізаторів для розробленої мови програмування.

Практичне заняття 18. Трансляція у ПОЛІЗ.
Ревю транслятора у ПОЛІЗ конструкцій розробленої мови програмування.

Практичне заняття 19. Трансляція у ПОЛІЗ та виконання ПОЛІЗ-програм
Ревю транслятора у ПОЛІЗ та розширення PSM.

Практичне заняття 20. Трансляція у ПОЛІЗ та виконання ПОЛІЗ-програм
Ревю транслятора у ПОЛІЗ та розширення PSM.

Практичне заняття 21. Генерування CIL-коду, асемблювання та виконання в CLR.
Ревю транслятора на CIL.

Практичне заняття 22. Представлення **КП 4**.
Представлення транслятора розробленої мови у postfix та на CIL.
Ревю транслятора розробленої мови у postfix та на CIL.

Практичне заняття 23. Генерування коду для JVM.
Ревю транслятора розробленої мови на Java-асемблер. Асемблювання та виконання програм.

Практичне заняття 24. ANTLR4
ANTLR4: Встановлення, використання. Перший сеанс.

Практичне заняття 25. Представлення **КП 4**.
Ревю прикладної програми з інтегрованим згенерованим парсером розробленої мови.

Практичне заняття 26. Представлення **РГР**.
Ревю прикладної програми з інтегрованим згенерованим парсером розробленої мови.

Практичне заняття 27. Розробка трансляторів.
Обговорення досвіду в контексті розробки і реалізації мов програмування.
Підсумки академічної діяльності.

Перелік робіт комп'ютерного практикуму

1. Специфікація мови програмування
2. Лексичний аналіз програм
3. Синтаксичний та семантичний аналіз програм
4. Трансляція та виконання програм
5. Побудова синтаксичного аналізатора засобами ANTLR4

Розрахунково-графічна робота

Індивідуальне завдання на проєктування та реалізацію простої імперативної мови програмування.

Контрольна робота

Метою контрольної роботи є закріплення та перевірка теоретичних знань із освітнього компонента, набуття студентами практичних навичок самостійного вирішення задач.

Модульна контрольна робота (МКР) виконується після вивчення Розділів 1-2. Кожен студент отримує завдання одного з 12 варіантів, відповідно до якого необхідно виконати

побудову граматики, скінченного або магазинного автомата, та виконати приведення граматики.

Самостійна робота студента

Теми для самостыйного опрацювання:

№ з/п	*Розділ*	*Тема*	*Джерела*	*Контроль*
1	1	Однопрохідний компілятор.	[2, розд. 1.2]	на практ. заняттях
2	1	Робота з граматиками у JFLAP.	[1, розд. 1.5; 15, розд. 3]	на практ. заняттях
3	1	Класифікація граматик Чомскі.	[1, розд. 1.3.4; 2, розд. 2.1.2]	на практ. заняттях
4	1	Усунення лівої рекурсії.	[1, розд. 1.3.5]	на МКР
5	1	Ebnf-visualizer (та інші інструменти) для генерування синтаксичних діаграм.	[1, розд. 1.2.4]	на практ. заняттях, РКП 1
6	1	Робота з регулярними виразами у JFLAP.	[15, розд. 4]	на практ. заняттях
7	1	Робота з скінченними автоматами у JFLAP.	[15, розд. 4]	на практ. заняттях, РКП 2, МКР
8	2	Лексичний аналіз методом розбору до роздільника.	[1, розд. 3.2]	на практ. заняттях,
9	2	Робота з МП-автоматами у JFLAP.	[1, розд. 1.3.4; 15, розд. 5]	на практ. заняттях, МКР
10	2	LL(k) і LR(k)– аналіз.	[4, розд. 4.4.3, 4.6.1]	на практ. заняттях, на екзамені
11	3	Використання ILASM і ILDASM.	[16; 17]	на практ. заняттях, екзамен, РКП 4
12	3	Використання java-асемблерів третіх сторін: Krakatau.	[18]	на практ. заняттях, РКП 4

Приблизний розподіл часу СРС

№ з/п	*Вид самостійної роботи*	*Кількість*	*Кількість годин СРС*
1	Виконання робіт комп'ютерного практикуму	5	30
2	Підготовка до РГР		6
3	Підготовка до практичних занять	27	27
4	Опрацювання тем, винесених на СРС	12	18
5	Підготовка до МКР		6
6	Підготовка до екзамену		30
		Разом	117

Політика та контроль

6. Політика навчальної дисципліни (освітнього компонента)

Система вимог при вивченні дисципліни:

- правила відвідування занять: заборонено оцінювати присутність або відсутність здобувача на аудиторному занятті, в тому числі нараховувати заохочувальні або штрафні бали. Відповідно до РСО даної дисципліни бали нараховують за відповідні види навчальної активності на лекційних та практичних заняттях.
- правила поведінки на заняттях: студент має можливість отримувати бали за відповідні види навчальної активності на лекційних та практичних заняттях, передбачені РСО дисципліни. Використання засобів зв'язку для пошуку інформації на гугл-диску викладача, в інтернеті, в дистанційному курсі на платформі Сікорський здійснюється за умови вказівки викладача;
- політика дедлайнів та перескладань: якщо студент не проходив або не з'явився на МКР (без поважної причини), його результат оцінюється у 0 балів. Перескладання результатів МКР не передбачено;
- політика щодо академічної доброчесності: Кодекс честі Національного технічного університету України «Київський політехнічний інститут» https://kpi.ua/files/honorcode.pdf встановлює загальні моральні принципи, правила етичної поведінки осіб та передбачає політику академічної доброчесності для осіб, що працюють і навчаються в університеті, якими вони мають керуватись у своїй діяльності, в тому числі при вивченні та складанні контрольних заходів з дисципліни «Основи розробки трансляторів»;
- при використанні цифрових засобів зв'язку з викладачем (мобільний зв'язок, електронна пошта, переписка на форумах та у соцмережах тощо) необхідно дотримуватись загальноприйнятих етичних норм, зокрема бути ввічливим та обмежувати спілкування робочим часом викладача.

7. Види контролю та рейтингова система оцінювання результатів навчання (РСО)

Система рейтингових (вагових) балів та критерії оцінювання

***Поточний контроль**: активність — питання і відповіді на лекційних заняттях, питання і відповіді та участь у ревю на практичних заняттях; вправи на практичних заняттях; МКР; виконання завдань до практичних занять; виконання та представлення робіт комп'ютерного практикуму, виконання та представлення РГР.*

***Календарний контроль**: провадиться двічі на семестр як моніторинг поточного стану виконання вимог силабусу ("атестація").*

Семестровий контроль: *екзамен.*

***Умови допуску до семестрового контролю**: виконані роботи 1-4 комп'ютерного практикуму, семестровий рейтинг — не менше 40 балів.*

Таблиця відповідності рейтингових балів оцінкам за університетською шкалою:

Бали	Оцінка
95 - 100	Відмінно
85 - 94	Дуже добре
75 - 84	Добре
65 - 74	Задовільно
60 - 64	Достатньо
Менше 60	Незадовільно
R < 40 або не виконана одна з робіт № 1-4 комп'ютерного практикуму	Не допущено

1) Активність

Активність (частота, змістовність) участі студента у процесі обговорення відповідних тем на заняттях — питання і відповіді на лекційних заняттях, питання і відповіді та участь у ревю на практичних заняттях, виконання вправ на практичних заняттях оцінюється, максимум, 10 балами, які може отримати кожен студент за семестр.

2) Роботи комп'ютерного практикуму

Максимальна кількість балів за усі виконані роботи комп'ютерного практикуму дорівнює 58 балів. Розподіл балів за виконання робіт практикуму:

№ з/п	Роботи комп'ютерного практикуму	Кількість балів
1	Специфікація мови програмування	8
2	Лексичний аналіз програм	14
3	Синтаксичний та семантичний аналіз програм	12
4	Трансляція та виконання програм	12
5	Побудова синтаксичного аналізатора засобами ANTLR4	12
Разом:		58

Критерії оцінювання:

Виконання робіт комп'ютерного практикуму:

- виконана у повному обсязі – максимальна кількість балів згідно номера роботи;
- виконана частково — відповідна частка від максимальної кількості балів.

3) Творче завдання

Студент може обрати додаткове завдання творчого характеру, результати виконання якого можуть бути опубліковані у наукових виданнях, або повідомлені на студентській науковій конференції з публікацією тез. Максимальна кількість балів за виконання творчого завдання — 30 балів, за умови, що загальна кількість балів не перевищує 100 балів.

4) Модульна контрольна робота

Максимальна кількість балів за модульну контрольну роботу дорівнює 10 балів.

Якість виконання роботи:

- виконана у повному обсязі з необхідними текстовими поясненнями дій та результатів – максимальна кількість балів;
- виконана частково з поясненнями — відповідна частка від максимальної кількості балів;
- виконана без текстових пояснень дій та результатів – не більше трьох балів;

5) Складання іспиту

Максимальний ваговий бал $r_{icп}$= 17

На усному екзамені студент обирає білет, який містить одне теоретичне питання та одне практичне завдання. Теоретичне питання оцінюється максимально до 5 балів, практичне — до 12 балів.

Умови позитивної проміжної атестації

Календарний контроль базується на поточній рейтинговій оцінці. Умовою позитивної атестації є значення поточного рейтингу студента не менше 50% від максимально можливого на час атестації. Бал, необхідний для отримання позитивного календарного контролю

доводиться до відома студентів викладачем не пізніше ніж за 2 тижні до початку календарного контролю.

Умови допуску до *іспиту*

Необхідною умовою допуску до іспиту є зарахування чотирьох робіт комп'ютерного практикуму та стартовий рейтинг (R_c) не менше 40 балів.

Розрахунок шкали (R) рейтингу:

Сума вагових балів контрольних заходів протягом семестру (шкала рейтингу) складає:

$$R = r_{КП} + r_{РГР} + r_{мКР} + r_{активн} + r_{icп} = 58 + 5 + 10 + 10+17= 100 \text{ балів.}$$

Максимальний стартовий рейтинг становить $R_c = r_{лек} + r_{практ} + r_{тв} + r_{мод} = 83$ бали.

Рейтинг іспиту дорівнює 17 балів. Мінімальний рейтинг допуску до іспиту становить 40 балів.

Таким чином, рейтингова шкала з кредитного модуля складає

$$R = 83+17=100 \text{ балів.}$$

Робочу програму навчальної дисципліни (силабус) «Основи розробки трансляторів»:

Склав доцент кафедри ІПЗЕ, к.т.н., доц. Стативка Юрій Іванович

Ухвалено кафедрою ІПЗЕ (протокол № 28 від 15.05.2023 р)

Погоджено Методичною комісією факультету[1] (протокол № 9 від 26.05.2023 р.)

[1] Методичною радою університету – для загальноуніверситетських дисциплін.