

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
ЧЕРКАСЬКИЙ ДЕРЖАВНИЙ ТЕХНОЛОГІЧНИЙ УНІВЕРСИТЕТ

«ЗАТВЕРДЖУЮ»

Голова приймальної комісії

Олег ГРИГОР

« 13 » 03 2022 р.

ПРОГРАМА

фахового іспиту

при вступі на навчання для здобуття освітнього ступеня магістра

зі спеціальності 123 – Комп'ютерна інженерія

(освітня програма – Системне програмування)

Черкаси 2022

I ПРОГРАМА ВСТУПНИХ ВИПРОБУВАНЬ

Програма фахового іспиту складена на підставі Порядку прийому на навчання для здобуття вищої освіти в 2022 році, затверджених Наказом Міністерства освіти і науки України 27 квітня 2022 року № 392, зареєстрованого в Міністерстві юстиції України 03 травня 2022 р. за №487/37823.

1.1 ВИМОГИ ДО РІВНЯ ПІДГОТОВКИ ВСТУПНИКІВ

До участі у конкурсі щодо зарахування на навчання для здобуття освітнього ступеня магістра зі спеціальності 123 – Комп'ютерна інженерія (освітня програма – Системне програмування) згідно переліку галузей знань і спеціальностей, за якими здійснюється підготовка здобувачів вищої освіти, затвердженого постановою Кабінету Міністрів України від 29 квітня 2015 року №266, допускаються особи, які здобули освітній ступінь бакалавра чи магістра або освітньо-кваліфікаційний рівень спеціаліста за спеціальностями згідно Додатку 5 Правил прийому до Черкаського державного технологічного університету в 2022 р. (на отримання ОС магістра).

Вступник має виявити базові знання з теорії та практики дисциплін, що виносяться на вступне випробування.

1.2 МЕТА ТА ЗАВДАННЯ ВСТУПНИХ ВИПРОБУВАНЬ

Перевірити відповідність знань, умінь, навичок вступників вимогам програм.

Оцінити ступінь підготовки вступників до закладів вищої освіти для навчання та здобуття ступеня магістра зі спеціальності 123 – Комп'ютерна інженерія (освітня програма – Системне програмування).

1.3 ПЕРЕЛІК ДИСЦИПЛІН ТА РОЗДІЛІВ З НИХ, ЯКІ ВИНОСЯТЬСЯ НА ВСТУПНІ ВИПРОБУВАННЯ

На іспит виносяться питання з навчальних програм наступних дисциплін: «Операційні системи», «Об'єктно-орієнтоване програмування»,

«Системне програмування для Windows», «Системне програмування», «Програмування».

Перелік тем з навчальних дисциплін, що виносяться на іспит:

1.3.1 Дисципліна «Операційні системи»

1. ОС, ЇХ ПРИЗНАЧЕННЯ, СКЛАДОВІ ЧАСТИНИ І ФУНКЦІЇ. ОС як інтерфейс між користувачем і ЕОМ. ОС як диспетчер ресурсів. Класифікація ОС.

2. ПОНЯТТЯ ПРОЦЕСУ ЯК МЕТОДУ КОНТРОЛЮ ПРОГРАМ, які виконує процесор, і керування ними. З складові процесу. Поняття віртуальної пам'яті. Віртуальна адреса, реальна адреса. Узагальнена модель ієрархічної ОС, її рівні.

3. ХАРАКТЕРИСТИКИ СУЧАСНИХ ОС І ЇХ ОСОБЛИВОСТІ. Основні особливості Windows (структурні рівні, модулі виконавчої системи та її структура в рамках моделі клієнт/сервер, типи користувацьких процесів). Традиційні та сучасні системи UNIX, їх короткі характеристики, характеристика Linux.

4. ПРОЦЕСИ І ПОТОКИ. Потоки в Windows. Керування процесами і потоками в Linux.

5. РОБОТА ОС ПО ЗАБЕЗПЕЧЕННЮ ПАРАЛЛЕЛЬНИХ ОБЧИСЛЕНЬ: взаємовиключення і багатозадачність. Принципи паралельних обчислень. Взаємовиключення: апаратний підхід (алгоритм Деккера, алгоритм Петерсона). Апаратна підтримка взаємовиключень в паралельних обчисленнях. Семафори. Монітори. Передача сповіщень. Принципи взаємного блокування. Запобігання взаємоблокуванню. Знищення взаємоблокувань. Алгоритм знаходження взаємоблокувань. Механізми паралельних обчислень в Windows.

6. ПАМ'ЯТЬ. КЕРУВАННЯ ПАМ'ЯТТЮ: вимоги до керування пам'яттю, розподілення пам'яті, сторінкова організація, сегментація. Пам'ять. Віртуальна пам'ять: апаратне забезпечення і керуючі структури. Програмне забезпечення операційної системи. Керування пам'яттю в UNIX та SOLARIS. Керування пам'яттю в Linux. Керування пам'яттю в Windows.

7. ПЛАНУВАННЯ. Планування в однопроцесорних системах: типи планування процесора, алгоритми планування, традиційне планування UNIX. Багатопроцесорне планування і планування реального часу. Планування в Linux.

1.3.2 Дисципліна «Об'єктно-орієнтоване програмування»

1. ОСНОВНІ ПРИНЦИПИ І ПОНЯТТЯ ОБ'ЄКТНО-ОРІЄНТОВАНОГО ПІДХОДУ.

Витоки ООП. Принципи об'єктно-орієнтованого підходу: інкапсуляція, наслідування, поліморфізм. Поняття об'єкту. Організація програми в об'єктно-орієнтованій моделі, повідомлення. Поняття класу. Порівняння процедурного та об'єктно-орієнтованого підходу.

2. ВВЕДЕННЯ ДО ПЛАТФОРМИ MICROSOFT .NET ТА МОВИ C#.

Основні поняття платформи Microsoft .NET та мови C#. Основи мови C#.

3 ОСНОВИ ВИКОРИСТАННЯ МОВИ XML ПІД ЧАС РОЗРОБКИ ДОДАТКІВ ДЛЯ .NET.

Основи використання мови XML під час розробки додатків для .Net

4. РЕАЛІЗАЦІЯ ГОЛОВНИХ КОНЦЕПЦІЙ ОБ'ЄКТНО-ОРІЄНТОВАНОГО ПРОГРАМУВАННЯ У МОВІ C#

Основні положення об'єктно-орієнтованого підходу. Класи та об'єкти, співвідношення між ними. Створення та руйнування об'єктів. Реалізація поліморфізму в C#

5. ОСНОВНІ БІБЛІОТЕКИ .NET

Принципи перевантаження операцій. Індикатори та властивості. Обробка виключень. Введення-виведення даних. Колекції. Рядки та регулярні вирази

6.ОСОБЛИВОСТІ ЗАСТОСУВАННЯ ПЛАТФОРМИ .NET ПРИ РОЗРОБЦІ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ.

Управління пам'яттю та вказівники. Атрибути. Збереження та відновлення стану об'єктів у .NET

1.3.3 Дисципліна «Системне програмування для Windows»

1. ОПЕРАЦІЙНІ СИСТЕМИ ТА ЇХ ІНТЕРФЕЙСИ.

Призначення ОС. Типи операційних систем. Інтерфейс програмування додатків Win32 API. Типи даних в Win32 API. Об'єкти та їх дескриптори в Windows.

2. УПРАВЛІННЯ ПОТОКАМИ ТА ПРОЦЕСАМИ В WINDOWS.

Визначення потоку. Контекст потоку. Стан потоку. Диспетчеризація та планування потоків. Створення та завершення потоків. Призупинка та відновлення потоків. Псевдодескриптори потоків. Обробка помилок в Windows. Визначення та створення процесу. Завершення процесів. Наслідування та дублювання дескрипторів. Псевдодескриптори процесів. Обслуговування потоків. Динамічна зміна пріоритетів потоків.

3. СИНХРОНІЗАЦІЯ ПОТОКІВ ТА ПРОЦЕСІВ.

Безперервні дії та команди. Поняття синхронізації. Програмна та апаратна реалізація синхронізації. Примітиви синхронізації. Синхронізація потоків в Windows. Критичні секції. Об'єкти синхронізації та функції очікування. М'ютекси. Події. Семафори. Взаємовиключний доступ до змінних. Атомарні операції. Заміна значення змінної. Умовна заміна значення змінної. Інкремент та декремент змінної. Зміна значення змінної. Глухі кути. Визначення та виявлення глухих кутів. Класифікація системних ресурсів. Відновлення заблокованого процесу. Попередження глухих кутів. Безпечно завершення потоків в Windows.

4. ПРОГРАМУВАННЯ КОНСОЛЬНИХ ДОДАТКІВ.

Структура консолі. Вхідний буфер консолі. Буфер екрану. Створення та вивільнення консолі. Стандартні дескриптори введення-виведення. Отримання дескриптору вікна консолі. Отримання і зміна заголовку консолі. Визначення максимального розміру вікна. Установка координат вікна. Створення та активація буферу екрану. Визначення та установка параметрів буферу екрану. Функції для роботи з курсором. Читання та установка атрибутів консолі. Введення-виведення високого рівня. Введення-виведення низького рівня. Режими введення-виведення консолі. Прокрутка буферу екрану.

5. ОБМІН ДАНИМИ МІЖ ПАРАЛЕЛЬНИМИ ПРОЦЕСАМИ.

Способи передачі даних між процесами. Зв'язки між процесами. Передача повідомлень. Синхронний та асинхронний обмін даними. Буферизація. Анонімні канали. Створення анонімних каналів. Організація керування віртуальною пам'яттю. Механізм відображення віртуальних адрес в реальні: сторінкова, сегментна та комбінована організація віртуальної пам'яті. Стратегії керування віртуальною пам'яттю: основні стратегії; локальність; робочі множини та підкачка сторінок.

6. ПРИЗНАЧЕННЯ ДИНАМІЧНИХ БІБЛІОТЕК.

Шаблон динамічної бібліотеки в MS VISUAL STUDIO. Використання функцій з динамічних бібліотек в MS VISUAL STUDIO. Способи приєднання DLL до програми в MS VISUAL STUDIO. Явне завантаження DLL. Ініціалізація і деініціалізація DLL в MS VISUAL STUDIO.

1.3.4 Дисципліна «Системне програмування»

1. КОМАНДИ. Формати машинних команд. Алгоритми виконання команд. Типи машинних команд. Команди пересилки даних. Арифметичні команди. Логічні команди і команди зсуву. Команди переходів і передач

керування. Команди обробки рядків і блоків даних. Команди асемблеру і псевдокоманди. Команди означення даних. Макрозасоби. Список команд i86/88, i286, i486.

2. ПРОГРАМИ. Структура програми. Завантаження сегментних адрес. Зв'язок сегментів усередині програми. Архітектура COM- і EXE- програми. Архітектура COM- програми. Архітектура EXE-програми- Префікс програмного сегмента. Спрощені способи запису програм.

3. СИСТЕМА ПЕРЕРИВАНЬ. Види переривань. Обробка переривання. Вектор переривання. Зразок програми, яка використовує переривання відеоадаптера для виведення рядків символів. Архітектура відеоадаптера. Види відеоадаптерів.

4. КЕРУВАННЯ КЛАВІАТУРОЮ. Рівні керування клавіатурою. Функції для роботи з клавіатурою. Таблиця скен-кодів і розширених кодів. Коди цифрової клавіатури. Призначення деяких кодів ASCII.

5. КЕРУВАННЯ ЕКРАНОМ. Функції керування екраном. Режими роботи екрана. Зображення символів на екрані дисплея. Відеопам'ять. Створення спеціальних символів на екрані.

6. ДИСКЕТА ТА ЖОРСТКИЙ ДИСК. Типи носіїв на гнучких дисках. Архітектура пристрою прямого доступу. Функції для роботи з дисками. Тест стану заданого диска.

7. КЕРУВАННЯ ФАЙЛАМИ. Розподіл пам'яті на диску. Розміщення файлів. Організація таблиці розміщення файлів. Каталог диска. Організація файлу. Підготовка до роботи з файлами. Перелік функцій, доступних через переривання 21H, які використовуються при керуванні файлами. Створення і вилучення файлу. Відкриття та закриття файлу. Функції керування файлом за методом дескриптора файлу. Керування каталогом.

8. КЕРУВАННЯ ПРИНТЕРОМ. Переривання і функції для керування принтером. Програмно-кероване виведення на друк. Опис команд керування принтером. Виведення спеціальних символів. Таймер і час.

9. РЕЗИДЕНТНІ ПРОГРАМИ. Структура TSR-програм. Основні функції переривань, що обслуговують резидентні програми.

10. МАТЕМАТИЧНИЙ СПІВПРОЦЕСОР. Команди математичного співпроцесора. Приклади застосування. Приклади дій з числами з плаваючою точкою.

11. МАНІПУЛЯТОР МИША. Керування маніпулятором миша з використанням функцій переривання 33h.

1.3.5 Дисципліна «Програмування»

1. АЛГОРИТМІЗАЦІЯ ЗАДАЧ. Блок-схеми алгоритмів. Лінійні, розгалужені, циклічні алгоритми.

2. СТРУКТУРА ПРОГРАМИ Pascal ABC. Базові типи даних.

3. ОПЕРАТОРИ МОВИ Pascal ABC. Стандартні процедури і функції мови Pascal ABC. Оператори розгалуження: безумовний та умовний перехід, конструкція типу CASE.

4. ПРОГРАМИ З ЦИКЛАМИ. Цикл з параметром (FOR .. TO .. DO ..). Цикл з передумовою (WHILE .. DO ..). Цикл з постумовою (REPEAT .. UNTIL ..). Вкладені цикли.

5. МАСИВИ. ЗАПИСИ І МНОЖИНИ. Одновимірні, двовимірні масиви, їх опис та обробка. Записи: опис та використання. Множини: опис та використання.

6. ПРОЦЕДУРИ І ФУНКЦІЇ. Звернення до підпрограм. Основні відомості про процедури, функції. Параметри процедур та функцій. Рекурсія. Ряди: їх обробка та використання.

7. ФАЙЛИ. Види файлів в Pascal ABC. Робота з файлами. Стандартні процедури для роботи з файлами. Вказівники та їх властивості. Лінійні списки, їх створення та перегляд.

1.4 СПИСОК РЕКОМЕНДОВАНОЇ ЛІТЕРАТУРИ

1.4.1 Операційні системи

1. Stallings, William. Operating systems: internals and design principles / William Stallings. – 7th ed. Prentice Hall, New Jersey, 2012, p.769. ISBN-13:978-0-13-230998-1.

2. Kusswurm Daniel. Modern X86 Assembly Language Programming/ Daniel Kusswurm. - Apress, 2019. — 604 p.

3. William Stallings. Operating Systems: Internals and Design Principles, 9th Edition. – Pearson, 2018. ISBN-10: 0-13-467095-7 | ISBN-13: 978-0-13-467095-9.

4. Операційні системи : навчальний посібник. [за ред. В. М. Рудницького] / І. М. Федотова-Півень, І. В. Миронець, О. Б. Півень, С. В. Сисоєнко, Т. В. Миронюк; Черкаський державний технологічний університет. – Харків : ТОВ «ДІСА ПЛІОС», 2019. – 216 с. ISBN 978-617-7645-93-0.

5. Richard Blum. Linux Command Line and Shell Scripting. Wiley; 3.

Edition. 2015. 816 p.

6. William Shotts. The Linux Command Line, 2nd Edition: A Complete Introduction. 2019. 504 p.

7. Jason Cannon. Linux for Beginners: An Introduction to the Linux Operating System and Command Line. 2013. 204 p.

1.4.2 Об'єктно-орієнтоване програмування

1. Об'єктно-орієнтоване програмування: конспект лекцій для студентів напряму підготовки "Комп'ютерні науки" всіх форм навчання / Ю. Е. Парфьонов, В. М. Федорченко, М. Ю. Лосєв, О. В. Щербаков. – Харків: Вид. ХНЕУ, 2010. – 312 с. (Укр. мов.)

2. Голуб Б. М. С#. Концепція та синтаксис: навч. посібник / Б. М. Голуб. – Львів : Видавничий центр ЛНУ імені Івана Франка, 2006. – 136 с.

3. Grady Booch, Robert A. Maksimchuk, et al. Object-Oriented Analysis and Design with Applications 3rd Edition. 2009. 720 p.

4. Troelsen A. Pro C# 2010 and the .NET 4 Platform, 5th Edition. Apress, 2010. 1752 p.

5. Trey Nash. Accelerated C# 2010 (Expert's Voice in C#), 1st ed. Edition, 2010. 656 p.

6. Harrison Ferrone. Learning C# by Developing Games with Unity 2019: Code in C# and build 3D games with Unity, 4th. 2019. 342 p

7. RB Whitaker. The C# Player's Guide (3rd Edition). Starbound Software. 2016. 406 p.

1.4.3 Системне програмування для Windows

1. Stallings, William. Operating systems: internals and design principles / William Stallings. – 7th ed. Prentice Hall, New Jersey, 2012, p.769. ISBN-13:978-0-13-230998-1

2. B. Johnson. Inside Visual Studio .NET 2003. Microsoft Press,U.S.; 2003. 550 p.

3. Julian Templeman, David Vitter. Visualstudio .Net: The .Net Framework Black Book (Black Book (Paraglyph Press)). 2001.816 p.

4. Bjorn Stroustrup. C++ High Performance: Master the art of optimizing the functioning of your C++ code, 2nd Edition 2nd ed. Edition.2018. 376 p.

5. Omprakash Tirunavukkarasu. Win32 API Reference in Visual Basic: Win32 API in Visual Basic . CreateSpace Independent Publishing Platform; 1st

edition. 2014. 420 p.

6. Johnson M.Hart Windows System Programming (4th ed.).2010. 610 p.

1.4.4 Системне програмування

1. IA-32 Intel® Architecture Software Developer's Manual. Vol. 2. Instruction Set Reference. Intel Corporation, 2002.

2. Daniel Kusswurm. Modern X86 Assembly Language Programming: Covers X86 64-Bit, Avx, Avx2, and Avx-512. 2019. 617 p.

3. Jeff Dunfemann. Assembly Language Step-by-Step. Programming with Linux. 2010. 646 p.

4. Kip R. Irvine. Assembly Language for x86 processors (7th edition). 2016. 873 p.

5. Randall Hyde. The Art of Assembly Language. 2010. 888 p.

6. Sivarama P. Dandamudi. Introduction to Assembly Language Programming For Pentium and RISC Processors, second edition. 2004. 702 p.

1.4.5 Дисципліна «Програмування»

1. Ковалюк Т.В. Основи програмування: Підручник для студентів – К., Вид. група ВНУ, 2005. – 384 с.

2. Шикова О.М. Основи програмування мовою Паскаль у прикладах і завданнях : Навчальний посібник / О.М. Шикова; МАУП. – Київ : МАУП, 2004. – 112с.

3. Белов Ю.А. Вступ до програмування мовою C++. / Ю.А. Белов, Т.О. Карнаух, Ю.В. Коваль, А.Б. Ставровський. – К.: Видавничо-поліграфічний центр «Київський університет», 2012. – 175 с.

4. Вступ до програмування мовою C++. Організація даних / Т. О. Карнаух, Ю. В. Коваль, М. В. Потієнко, А. Б. Ставровський. – К.: ВПЦ "Київський університет", 2015.

5. Я. М. Глинський, В. Є. Анохін, В. А. Ряжська. C ++ і C++Builder. Навч. посіб.- 3-тє вид. Львів : СПД Глинський, 2003. – 193 с.

2 ПОЯСНЮВАЛЬНА ЗАПИСКА ДО ВСТУПНИХ ВИПРОБУВАНЬ

Вимоги до фахового іспиту відповідають вимогам чинних навчальних програм згідно стандарту вищої освіти зі спеціальності 123 – **Комп'ютерна інженерія (освітня програма – Системне програмування)**.

Час тестування – 2 астрономічні години (120 хвилин).

Вступні випробування проводяться у формі тестування в письмовій формі.

Тестове завдання містить 15 пунктів і складається з двох блоків.

Блок 1 – 11 завдань (завдання 1-10, 13). *Блок 2* – 4 завдання (завдання 11, 12, 14, 15).

Блок 1 містить завдання закритого типу, *Блок 2* – відкритого типу.

Для тестового *Блоку 1* подано від 3 до 5 варіантів відповідей, з яких тільки одна правильна. Тестове питання вважається виконаним правильно, якщо вступник вказав саме правильну відповідь.

Блок 2 містить 3 завдання практичного типу і 1 завдання теоретичного типу.

Правильність виконання завдань оцінюється відповідно до критеріїв оцінювання знань.

Перевіряючий не зобов'язаний читати розв'язання завдань, що наведені вступником у чернетці.

Оцінка за письмову роботу виставляється як сума балів за кожне завдання і являє собою сумарний рейтинг.

Результати фахового іспиту оцінюються за шкалою від 100 до 200 балів.

Особи, які набрали на вступних випробуваннях менше ніж 125 бали, позбавляються права участі в конкурсі за спеціальністю (освітньою програмою).

3 КРИТЕРІЇ ОЦІНЮВАННЯ ВСТУПНИХ ВИПРОБУВАНЬ

1. Результати фахового іспиту оцінюються за шкалою від 100 до 200 балів і є результатом додавання до 100 балів суми балів, отриманих за виконання кожного завдання тесту.

Правила виконання завдань відповідних типів та вимог до запису відповідей вступників наведено в екзаменаційних білетах (тестових завданнях).

Тестове завдання містить 15 пунктів і складається з двох блоків.

Блок 1 – 11 завдань закритого типу (завдання 1-10, 13).

Блок 2 – 4 завдання відкритого типу (завдання 11, 12, 14, 15).

Правильна відповідь на кожне тестове питання 1-10 оцінюється у 4 бали, неправильна – 0 балів. Максимальна кількість балів, яку абітурієнт може одержати за відповіді на тести 1-10, складає 40 балів. Вибрану відповідь на тестове питання пояснювати не потрібно.

Правильна відповідь на кожне завдання 11-13 оцінюється у 10 балів, неправильна – 0 балів. Максимальна кількість балів, яку абітурієнт може одержати за відповіді на завдання 11-13, складає 30 балів. Вибрану відповідь на тестове питання 13 пояснювати не потрібно.

Правильний розв'язок задачі 14 оцінюється у 10 балів, повна правильна відповідь на теоретичне питання 15 оцінюється у 20 балів.

Якщо відповідь на завдання абітурієнта містить помилки, його оцінка знижується:

1. У відповіді на тести 1-10:

- є виправлення – знімається 2 бали;

- відповідь не позначено або в відповіді на тестове питання відмічено кілька відповідей відразу – знімається 4 бали.

2. У відповіді для задач 11,12:

- висновок про дію фрагмента програми повністю невірний – знімається 10 балів;

- правильно описано послідовність дій операторів у фрагменті програми, але зроблено неправильний або дуже розпливчастий висновок про дію фрагмента в цілому – знімається 6 балів;

- є мілкі неточності в описі елементів фрагменту програми, але в цілому формулювання дії фрагменту вірне – знімається 3 бали;

- немає опису дій елементів програми і є розпливчастий, неконкретний висновок – знімається 8 балів.

3. У відповіді на тест 13:

- є виправлення – знімається 5 балів;

- відповідь не позначено або в відповіді на тестове питання відмічено кілька відповідей відразу – знімається 10 балів.

4. У відповіді на задачу 14:

- невірний результат із-за технічної помилки в кінці розв'язку – знімається 3 балів;

- невірний результат внаслідок технічної помилки на початку розв'язку – знімається 4 балів;

- правильний хід розв'язку, але відсутній результат – знімається 5 балів;

- правильний підхід до розв'язання задачі, але відсутнє знання технічних елементів процесу розв'язання – знімається 6 балів;

- правильний опис процесу розв'язання, але сам розв'язок відсутній – знімається 7 балів;

- є знання основних елементів процесу розв'язання, але розв'язок відсутній – знімається 8 балів.

5. У відповіді на теоретичне питання 15:

- якщо не розкрито питання повністю, знімається 20 балів;

- якщо відповідь розпливчаста, неконкретна, або викладено інформацію, яка мало стосується питання, оцінка знижується на 16 балів;

- якщо відповідь вірна лише на третину, оцінка знижується на 14 балів;

- якщо відповідь вірна лише на половину, оцінка знижується на 10 балів;

- якщо відповідь містить 3-4 незначні неточності, оцінка знижується на 5 балів;

- якщо відповідь містить 1-2 незначні неточності, оцінка знижується на 2 бали;

Оцінка за тест виставляється як сума балів за кожне завдання.

2. Особи, які набрали на вступних випробуваннях менше ніж 125 бали, позбавляються права участі в конкурсі за спеціальністю (освітньою програмою).

Голова фахової атестаційної комісії
зі спеціальності 123 – Комп'ютерна інженерія
(освітня програма –
Системне програмування)



д.т.н., доцент Віра БАБЕНКО