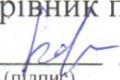


Міністерство освіти і науки України
Національний аерокосмічний університет ім. М.Є. Жуковського
«Харківський авіаційний інститут»

Кафедра математичного моделювання та штучного інтелекту (№ 304)
(назва кафедри)

ЗАТВЕРДЖУЮ

Керівник проектної групи

 Олексій КАРТАШОВ
(підпис) (ініціали та прізвище)

« 31 » 08 2021 р.

РОБОЧА ПРОГРАМА ОBOB'ЯЗКОВОЇ НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Архітектура обчислювальних систем
(назва навчальної дисципліни)

Галузі знань: 11 "Математика та статистика"
(шифр і найменування галузі знань)

Спеціальність: 113 "Прикладна математика"
(код та найменування напрямку підготовки)

Освітня програма: "Обчислювальний інтелект"
(найменування освітньої програми)

Форма навчання: денна

Рівень вищої освіти: перший (бакалаврський)

Харків 2021 рік

Робоча програма Архітектура обчислювальних систем
(назва дисципліни)
для студентів за спеціальностями 113 "Прикладна математика",
освітніми програмами "Обчислювальний інтелект"
« 27 » серпня 2021 р., – 12 с.

Розробник: Меняйлов Є.С., старший викладач каф.304
(прізвище та ініціали, посада, наукова ступінь та вчене звання)



(підпис)

Робочу програму розглянуто на засіданні кафедри математичного
моделювання та штучного інтелекту

(назва кафедри)
Протокол № 2 від « 27 » серпня 2021 р.

Завідувач кафедри Д.Т.Н., доц.
(наукова ступінь та вчене звання)



(підпис)

Андрій ЧУХРАЙ
(ініціали та прізвище)

1. Опис навчальної дисципліни

Найменування показників	Галузь знань, спеціальність, освітня програма, рівень вищої освіти	Характеристика навчальної дисципліни
		Денна форма навчання
Кількість кредитів – 5	Галузь знань <u>11 "Математика та статистика"</u> (шифр та найменування) Спеціальність <u>113 "Прикладна математика"</u> (код та найменування) Освітня програма <u>"Обчислювальний інтелект"</u> (найменування) Рівень вищої освіти: перший (бакалаврський)	Обов’язкова
Кількість модулів – 2		Навчальний рік
Кількість змістових модулів – 2		2021/2022
Індивідуальне завдання – не має		Семестр
Загальна кількість годин: денна – 64/150		4-й
		Лекції
Кількість тижневих годин для денної форми навчання: аудиторних – 4 самостійної роботи студента – 5,4		32 годин
		Практичні, семінарські
		32 годин
		Лабораторні
		32 годин
		Самостійна робота
		86 години
		Вид контролю
		модульний контроль, іспит

Примітка

Співвідношення кількості годин аудиторних занять до самостійної і індивідуальної роботи становить:

для денної форми навчання – 64/86.

2. Мета та завдання навчальної дисципліни

Мета: дати знання про сучасні методи схемотехніки та створення архітектури комп'ютерних систем для завдань проектування.

Завдання: вивчення схемотехнічних та структурних рішень для створення сучасних архітектур комп'ютерних систем.

Формування компетентностей:

ЗКЗ – Знання та розуміння предметної області та професійної діяльності.

ФК5 – Базові знання в області комп'ютерної інженерії в обсязі, необхідному для розуміння базових принципів організації та функціонування апаратних засобів сучасних систем обробки інформації, основних характеристик, можливостей і областей застосування обчислювальних систем різного призначення.

Результати навчання:

ПРН23 – Знання організаційних, технічних, алгоритмічних та інших методів і засобів захисту комп'ютерної інформації, законодавчих актів і стандартів для цієї області, сучасних криптосистем; уміння їх застосовувати в процесі професійної діяльності.

Міждисциплінарні зв'язки: програмування, об'єктно орієнтоване програмування, теорія алгоритмів та математична логіка, операційні системи, проектування програмних систем, розподілені обчислення.

2. Тематичний план навчальної дисципліни

Змістовний модуль 1.

Поняття про архітектуру обчислювальних систем. Кодування інформації в обчислювальних машинах.

Тема 1. Системи числення.

Принципи фон Неймана (як парадигма сучасних ЕОМ). Вплив основних класів задач на структуру (архітектуру) комп'ютерів. Організація шин. Шина адресу та даних. Правила переведення чисел з однієї системи числення до іншої.

Особливості використання систем числення з основою, що складає ступінь двійки. Зображення у пам'яті комп'ютера позитивних цілих чисел та додаткові двійкові коди негативних чисел. Обґрунтування доцільності використання представлення порядків чисел з плаваючою точкою у формі характеристики.

Тема 2. Кодування інформації в обчислювальних машинах.

Внутрішнє уявлення чисел, символів і зображень у пам'яті обчислювальних машин. Поняття: архітектури, структура та схематехніка ЕОМ. Абстракція функцій, даних. Необхідність у мікромінітюаризації процесорів. Закон Мура масштабування апаратного забезпечення обчислювальних систем. Зв'язок тактової частоти цифрового автомата та виділяемого ним тепла. Схематехнічні наслідки. Особливості сучасного техпроцесу великих інтегральних мікросхем та його вплив на архітектуру комп'ютерів. Абстракція центрального процесора.

Тема 3. Виконання розрядових логічних операцій.

Алгоритми розрядових логічних операцій. Алгоритми двійкового складання, множення та ділення. Зображення у пам'яті комп'ютера чисел з плаваючою крапкою. Принципи кодування у пам'яті комп'ютера символів клавіатури. Додавання та віднімання у додатковому коді чисел з фіксованою точкою. Числа з плаваючою точкою: характеристика, нормалізована мантиса. Стандарт *IEEE 754*. Числа з плаваючою точкою: реалізація додавання та віднімання, важливість дій з розширеною мантисою, рекомендації стандарту *IEEE 754* щодо розширення, ненормалізовані представлення.

Тема 4. Основи програмування мовою Асемблер.

Мова програмування асемблер, як мова програмування низького рівня. Основні типи даних. Регістри. Структура регістрів акумуляторів процесору. Регістри стеку та команд. Пам'ять. Стек. Логічні операції в мові асемблер. Основні команди асемблера для процесорів Intel. Команди пересилання даних. Пряма і непряма адресація. Зміщення. Циклічне зміщення. Арифметичні команди у мові асемблер. Різниця між 16 та 32 бітними командами. Команди поширення знаку. Команди порівняння. Команди умовного і безумовного переходу.

Змістовний модуль 2.

Спільне використання мов високого рівня та мови Асемблер.

Тема 5. Реалізація складних логічних структур мовою Асемблер.

Реалізація логічних структур мов високого рівня на мові асемблер. Оператор *if*. Цикл по убуванню. Вкладені цикли по убуванню, коли значення змінних невідоме. Цикл по зростанню. Вкладені цикли по зростанню з відомими значеннями змінних. Вкладені цикли по зростанню з невідомими значеннями змінних.

Тема 6. Етапи трансляції, компіляції, зв'язування програм.

Етапи трансляції, компіляції, зв'язування програм. Основи машинних кодів. Аналіз результатів дизасемблювання програм. PE-формат.

Змінна. Розміщення змінних. Локальні та глобальні змінні. Секції даних. Масиви. Розміщення багатомірних масивів у пам'яті. Вказівник. Динамічна пам'ять. Виклик функцій. Угоди про виклик. Робота з динамічною пам'яттю. Структурне програмування на мові асемблер. Робота з делегатами. Об'єктно-орієнтоване програмування на асемблері. Виклики функцій. Угоди __stdcall, __cdecl. Робота з файлами. Робота з динамічною пам'яттю. Функції роботи з системною купою.

3. Структура навчальної дисципліни

Назви розділів і тем	Кількість годин											
	денна форма						заочна форма					
	усього	у тому числі					усього	у тому числі				
		л	п	лаб.	інд.	с. р.		л	п	лаб.	інд.	с. р.
1	2	3	4	5	6	7	8	9	10	11	12	13
Змістовний модуль 1												
Тема 1. Системи числення.	14	2	2			10						
Тема 2. Кодування інформації в обчислювальних машинах.	12		2			10						
Тема 3. Виконання розрядових логічних операцій.	16	4	2			10						
Тема 4. Основи програмування мовою Асемблер.	46	10	12			24						
Разом за розділом 2	90	16	18			54						
Змістовний модуль 2												
Тема 5. Реалізація складних логічних структур мовою Асемблер.	31	8	8			15						
Тема 6. Етапи трансляції, компіляції, зв'язування програм	29	8	6			15						
Разом за розділом 3	60	16	14			30						
Усього годин	150	16	32			86						

5. Теми семінарських занять

№ з/п	Назва теми	Кількість годин
1		
2		
	Разом	

6. Теми практичних занять

№ з/п	Назва теми	Кількість годин
1		
2		
	Разом	

7. Теми лабораторних занять

№ з/п	Назва теми	Кількість годин
1	Перевід чисел з одної системи числення до іншої, в тому числі використання систем числення з основою, що складає ступінь двійки.	2
2	Кодування інформації в обчислювальних машинах.	2
3	Виконання розрядових логічних операцій та побудова додаткових двійкових кодів від'ємних чисел.	2
4	Використання масок та логічних операцій у мові Асемблер.	2
5	Виконання арифметичних операцій. Особливості використання у мові Асемблер.	2
6	Обчислення арифметичних виразів мовою Асемблер.	2
7	Обробка бітових рядків у мові Асемблер.	2
8	Робота з арифметичними зсувами у мові Асемблер.	2
9	Робота з циклічними зсувами у мові Асемблер.	2
10	Організація обробки рядків.	2
11	Динамічна пам'ять. Організація циклів. Одновимірні масиви	2
12	Динамічна пам'ять. Організація циклів. Багатовимірні масиви.	4
13	Виклик функцій і передача параметрів у функцію мовою Асемблера.	4
14	Робота з файлами	2
	Разом	32

8. Завдання для самостійної роботи

№ з/п	Види, зміст самостійної роботи	Кількість годин
1	Індивідуальне завдання на тему: «Вивчити структуру процесора. Ознайомитися з основними командами».	5
2	Індивідуальне завдання на тему: «Вивчити структуру підсистеми пам'яті та її характеристики».	5
3	Індивідуальне завдання на тему: «Ознайомитися з класами запам'ятовуючих пристроїв»	5
4	Індивідуальне завдання на тему: «Вивчити структуру оперативного пристрою та постійного запам'ятовуючого пристрою».	5

1	2	3
5	Індивідуальне завдання на тему: «Ознайомитися з основними елементами та принципами роботи стекової пам'яті».	5
6	Індивідуальне завдання на тему: «Ознайомитися з поняттям та структурою асоціативних пристроїв, що запам'ятовують. Вивчити структуру кеш-пам'яті».	5
7	Індивідуальне завдання на тему: «Ознайомитися з структурою підсистеми управління. Вивчити основні функції і склад пристрою управління».	5
8	Індивідуальне завдання на тему: «Ознайомитися з принципами функціонування пристрою управління».	5
9	Індивідуальне завдання на тему: «Ознайомитися з принципами управління введенням-виведенням. Ознайомитися з поняттям та структурою системної магістралі ("загальна шина") та контролерів зовнішніх пристроїв»	5
10	Індивідуальне завдання на тему: «Ознайомитися з принципами роботи програмно-керованої передачі інформації».	5
11	Індивідуальне завдання на тему: «Вивчити основні принципи організації переривань. Ознайомитися з поняттям векторів переривань».	5
12	Індивідуальне завдання на тему: «Ознайомитися з типами переривань та їх програмною реалізацією».	5
13	Індивідуальне завдання на тему: «Ознайомитися з типами периферійних пристроїв».	6
14	Індивідуальне завдання на тему: «Запам'ятати регістри: співпроцесора x87, розширення MMX, блоку XMM».	5
15	Індивідуальне завдання на тему: «Вивчити програмні й апаратні порти. Ознайомитися з класифікацією процесорів за складністю команд. CISC, RISC, VLIW процесори».	6
16	Індивідуальне завдання на тему: «Ознайомитися з організацією та принципами побудови багатоядерних архітектур. Ознайомитися з поняттям віртуальних ядер та їх програмною реалізацією».	4
17	Індивідуальне завдання на тему: «Ознайомитися з організацією та принципами побудови ARM архітектури».	5
	Разом	86

9. Індивідуальні завдання

Не має.

10. Методи навчання

1. Пояснювально-ілюстративний (інформаційно-рецептивний) метод: та метод проблемного виконання (лекція).
2. Репродуктивний (лабораторні роботи).
3. Частково-пошуковий (евристичний) та дослідницький: (самостійна робота та виконання РР).

11. Методи контролю

Визначення рівня засвоєння студентом навчального матеріалу дисципліни здійснюється шляхом проведення поточних і підсумкових контролів. У завдання поточного контролю входить систематична перевірка розуміння й засвоєння студентом програмного матеріалу, виконання практичних робіт, уміння самостійно проробляти тексти складання конспектів, здатності усно або письмово представляти певний матеріал.

Перед модульним контролем ставиться завдання перевірки глибини засвоєння студентом програмного матеріалу дисципліни, логіки й взаємозв'язки між її окремими розділами, здатності творчо використати придбані знання, уміння сформулювати своє відношення до проблеми, що впливає зі змісту дисципліни.

12. Критерії оцінювання та розподіл балів, які отримують студенти

12.1. Розподіл балів, які отримують студенти (кількісні критерії оцінювання для одержання позитивної оцінки)

Складові навчальної роботи	Бали за одне заняття (завдання)	Кількість занять (завдань)	Сумарна кількість балів
Змістовний модуль 1			
Виконання і захист лабораторних робіт	3...5	7	21...35
Модульний контроль	0...15	1	0...15
Змістовний модуль 2			
Виконання і захист лабораторних робіт	3...5	7	21...35
Модульний контроль	0...15	1	0...15
Усього за семестр			0...100

Семестровий контроль іспит проводиться у разі відмови студента від балів поточного тестування й за наявності допуску до іспиту. Під час складання семестрового іспиту студент має можливість отримати максимум 100 балів.

Білет для іспиту складається з

	Теоретичні питання														
Номер питання	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Кількість балів	1	4	4	5	2	9	1	2	12	5	5	5	5	5	5
Всього	70														

	Практичні питання	
Номер питання	1	2
Кількість балів	15	15
Всього	30	

12.2. Якісні критерії оцінювання

Необхідний обсяг знань для одержання позитивної оцінки (у тому числі використовуючи готові програмні рішення та спеціалізовані програмні бібліотеки):

- базові компоненти архітектури обчислювальних систем;
- базові поняття мови програмування Асемблер;
- основні принципи роботи з даними.

Необхідний обсяг вмінь для одержання позитивної оцінки (у тому числі використовуючи готові програмні рішення та спеціалізовані програмні бібліотеки):

- застосовувати на практиці мову Асемблер для обробки даних;
- використовувати особливості Архітектури обчислювальних систем для вирішення практичних завдань.

12.3 Критерії оцінювання роботи студента протягом семестру

Задовільно (60-74). Мати мінімум знань та умінь. Відпрацювати та захистити всі лабораторні роботи. Вміти використовуючи готові програмні рішення у вирішенні практичних завдань.

Добре (75 - 89). Твердо знати мінімум знань, виконати усі завдання. Показати вміння виконувати та захищати всі лабораторні роботи в обумовлений викладачем строк з обґрунтуванням рішень та заходів, які запропоновано у роботах. Вміти використовувати особливості архітектури обчислювальних систем для вирішення практичних завдань мовою Асемблер.

Відмінно (90 - 100). Повно знати основний та додатковий матеріал. Знати усі теми. Орієнтуватися у підручниках та посібниках. Вміти самостійно створювати програмні рішення для реалізації завдань в лабораторних роботах і розрахунковій роботі.

Шкала оцінювання: бальна і традиційна

Сума балів	Оцінка за традиційною шкалою	
	Іспит, диференційований залік	Залік
90 – 100	Відмінно	Зараховано
75 – 89	Добре	
60 – 74	Задовільно	
0 – 59	Незадовільно	Не зараховано

13. Методичне забезпечення

Навчально-методичний комплекс дисципліни, який включає в себе:

- робоча програма дисципліни;

– конспект лекцій, підручники (навчальні посібники), в тому числі в електронному вигляді, які за змістом повністю відповідають робочій програмі дисципліни;

– методичні вказівки та рекомендації для виконання курсових робіт та проектів, розрахункових та розрахунково-графічних робіт, лабораторних та практичних робіт, а також рекомендації для самостійної підготовки;

– тематики індивідуальних завдань;

– приклади розв’язування типових задач чи виконання типових завдань;

– питання, тести для контрольних заходів;

– каталоги інформаційних ресурсів.

14. Рекомендована література

Основна література

1. Калиш Г.Г. Основы вычислительной техники // Высшая школа, 2000. – 271 с.
2. Мураховский В.И. Устройство компьютера // АСТ-Пресс Книга, 2003. – 639 с.
3. А. Г. Фролов, Г. В. Фролов. Аппаратное обеспечение IBM PC // «Диалог-МИФИ», 1992. – 208 с.
4. Владислав Пирогов. Ассемблер и дизассемблирование. – СПб.: БХВ-Петербург, 2006. – 464 с.
5. Абель П. Язык Ассемблера для IBM PC и программирования. – М.: Высшая школа, 1992. – 447 с.
6. Брэдли Д. Программирование на языке ассемблера для персональной ЭВМ фирмы IBM.– М.: Радио и связь, 1988. – 448 с.
7. Галисеев Г.В. Ассемблер IBM PC. Самоучитель.: – М.: Издательский дом «Вильямс», 2004. – 304 с.: ил.
8. Дао Л. Программирование микропроцессора 8088. – М.: Мир, 1988. – 357 с.
9. Жуков А.В., Авдюхин А.А. Ассемблер. – СПб.: БХВ-Петербург, 2003. – 448 с.: ил.
10. Зубков С.В., Ассемблер для DOS, Windows и UNIX. – М.: ДМК Пресс, 2000. – 608 с.: ил. (Серия «Для программистов»).
11. Ирвин К. Язык ассемблера для процессоров Intel, 4-е издание.: пер. с англ. – М.: Издательский дом «Вильямс», 2005. – 912 с.: ил. – Парал. тит. англ.(см. также свежее [7-ое издание](#) в оригинале)
12. Нортон П., Соухэ Д. Язык ассемблера для IBM PC.– М.: Компьютер, 1992.– 352 с.
13. Пильщиков В.Н. Программирование на языке ассемблера IBM PC.– М.: ДИАЛОГ-МИФИ, 1994–2014 288 с.

Допоміжна література

1. Intel ® 64 and IA-32 Architectures Optimization Reference Manual
2. Intel ® 64 and IA-32 Architectures Software Developer's Manual Volume 1: Basic Architecture
3. Intel ® 64 and IA-32 Architectures Software Developer's Manual Volume 2A: Instruction Set Reference, A-M
4. Intel ® 64 and IA-32 Architectures Software Developer's Manual Volume 2B: Instruction Set Reference, N-Z
5. Intel ® 64 and IA-32 Architectures Software Developer's Manual Volume 3A: System Programming Guide, Part 1
6. Intel ® 64 and IA-32 Architectures Software Developer's Manual Volume 3B: System Programming Guide, Part 2
7. Leiteman J.C. 32/64-BIT 80x86 Assembly Language Architecture. © 2005, Wordware Publishing, Inc (568 pages) 2320 Los Rios Boulevard Plano, Texas 75074
8. Turbo Assembler® Version 3.2 User's Guide Borland International. Inc 1800 Green Hills Road P.O. BOX 660001, Scotts Valley, CA 95067-0001
9. Лямин Л.В. Макроасемблер MASM.— М.: Радио и связь, 1994.— 320 с.: ил.
10. Магда Ю. Ассемблер для процессоров Intel Pentium. — Спб.: Питер, 2006. — 410 с.: ил.
11. Юров В. Assembler: специальный справочник.— Спб.: Питер, 2001.— 496 с.: ил.
12. Юров В. Assembler. Практикум. 2-е изд. — Спб.: Питер, 2006. — 399 с.: ил.
13. Юров В. Assembler. Учебник для вузов. 2-е изд. — Спб.: Питер, 2007. — 637 с.: ил.

15. Посилання на інформаційні ресурси в Інтернеті, відео-лекції, інше методичне забезпечення

x86asm.net – Assembly x86 Manual Guide
wasm.ru - WASM SDK
www.masm32.com/ — MASM32 SDK
flatassembler.net/ — flat assembler (FASM)
www.nasm.us/ — The Netwide Assembler (NASM)