

Відокремлений структурний підрозділ
«Конотопський індустріально-педагогічний фаховий коледж СумДУ»
Циклова комісія комп'ютерних та математичних дисциплін



“ЗАТВЕРДЖУЮ”

Заступник директора
з навчальної роботи

Кравченко О.О.

2023 року

РОБОЧА ПРОГРАМА НАВЧАЛЬНОЇ ДИСЦИПЛІНИ

Програмування (Основи програмування та алгоритмічні мови)

(назва навчальної дисципліни)

Галузь знань 01 Освіта/Педагогіка

(шифр і назва напрямку підготовки)

спеціальність 015 Професійна освіта

(шифр і назва спеціальності)

Спеціалізація 39 Цифрові технології

відділення Професійної освіти

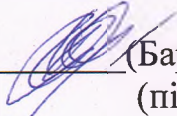
(назва відділення)

Робоча програма «Програмування. (Основи програмування та алгоритмічні мови)» для студентів за галуззю знань за галуззю знань 01 Освіта/Педагогіка спеціальність 015 Професійна освіта
«30» серпня 2023 року - 19 с.

Розробники: викладач ВСП «Конотопський індустріально-педагогічного фахового коледжу СумДУ Бараболіков В.М..

Робоча програма затверджена на засіданні циклової комісії Комп'ютерних та математичних дисциплін

Протокол від. «30_» серпня 2023 року № 1_

Голова циклової комісії  (Бараболіков В.М.)
(підпис)

«__» серпня 2023 року

Схвалено методичною радою ВСП «Конотопського індустріально-педагогічного фахового коледжу СумДУ.

Протокол від «__» _____ 20__ року № ____

«__» _____ 20__ року Голова _____ (Кравченко О. О.)
(підпис)

1. Опис навчальної дисципліни

Найменування показників	Галузь знань, напрям підготовки, освітньо-кваліфікаційний рівень	Характеристика освітнього компоненту		
		денна форма навчання		
Кількість кредитів-3	Галузь знань 01 «Освіта» (шифр і назва)	Нормативна		
	Напрямок підготовки (шифр і назва)			
Розділів – 2	Спеціальність 015 Професійна освіта	Рік підготовки:		
Тем – 18		2-й	3-й	
Індивідуальне науково-дослідне завдання – <u>курсова робота</u>		Семестр		
Загальна кількість годин - 90				
Годин для денної форми навчання: аудиторних – 64 самостійної роботи студента - 26		Освітньо-професійний ступінь: Фаховий молодший бакалавр з Професійної освіти	Лекції	
			40	
	Практичні			
			8	
	Лабораторні			
			16	
	Самостійна робота			
	26			
Індивідуальні завдання – курсова робота - 20 год.				
Вид контролю: залік				

Примітка.

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

для денної форми навчання – **1 : 0,9**

для заочної форми навчання –

2. Мета та завдання навчальної дисципліни

Мета та завдання дисципліни, її місце в навчальному процесі.

Мета дисципліни:

дати майбутнім спеціалістам належні знання з теоретичних основ алгоритмів, та мови програмування C ++, C#, принципів побудови основних алгоритмічних структур розв'язку типових задач, алгоритмізації та програмування основних виробничих задач.

Основні завдання курсу:

ознайомлення з поняттям алгоритму та його властивостями, способами та формами подання алгоритму, основними базовими структурами алгоритмів;

формування знань правил запису структурованих алгоритмів і програм, принципів побудови опису програми на мовах , C ++, C#; основних понять мови: числа, рядки, описи, ідентифікатори, оператори, величини, операції, типи даних та набір функцій і операцій доступних для кожного з типів даних; призначень та правил описування вказівок розгалуження й повторювання;

формування вмінь та навичок складання алгоритмів і програм; створення алгоритмів-процедур та алгоритмів-функцій; складання програм для розв'язку математичних, фізичних та економічних задач.

Програмні результати навчання

- принципи складання алгоритмів та складання блок-схем до алгоритмів;
 - оператори вводу-виводу інформації правила їх використання;
 - умовні оператори, оператори вибору варіанту та їх можливості при створенні програмних продуктів;
 - можливості при створенні циклів на мовах , C ++, C#; їх особливості їх використання;
 - правила написання функцій процедур та класів у , C ++, C#;
 - організацію роботи з файлами, організацію доступу до зовнішніх носіїв;
 - використання динамічної пам'яті;
 - графічні оператори, правила побудови растрових та розширених зображень;
 - інтерфейс програмування C#, C# , основні компоненти сторінок компонент, їх основні властивості та правила їх використання;
 - роботу з базами даних в C#, C# ;
 - методика написання довідки та створення інсталяційних пакетів.
- Здатність спілкуватися іноземною мовою
 - Знання та розуміння предметної області та розуміння професійної діяльності.
 - Здатність алгоритмічно та логічно мислити.

- Здатність розробляти модулі і компоненти програмного забезпечення за допомогою типових алгоритмів та інструментів.
- Здатність реалізовувати всі етапи життєвого циклу програмного забезпечення.
- Здатність створювати програмні продукти з використанням хмарних технологій

Місце освітнього компоненту в структурно-логічній схемі ОПП та міжпредметні зв'язки:

Із курсом “Програмування” студенти ознайомлюються після вивчення наступних професійно-орієнтованих дисциплін: інформатика; дискретна математика, архітектура комп'ютера. Нижче подано перелік дисциплін, що можуть читатися після курсу “Програмування”, певним чином залежні від нього або перетинаються із ним: Операційні системи; Архітектура та проектування програмного забезпечення; Аналіз вимог до програмного забезпечення; Якість програмного забезпечення; Моделювання програмного забезпечення; Безпека програм та даних.

3. Програма навчальної дисципліни

Розділ 1. Мова програмування C++

Тема 1.1 Основні поняття

Загальні відомості про мову. Алфавіт мови. Структура програми. Класифікація операцій та даних. Основні стандартні типи даних. Операції над стандартними типами даних. Стандартні функції.

Література: [1] [1] с. 20 – 25

[2] с.151-164

Тема 1.2 Потоки. Введення - виведення даних.

Програмування вводу виводу у мові. Керуючі структури мови. Послідовне виконання. Складений оператор.

Література: [1] с.67-86,

[2] с.84-114

Лабораторне заняття - 1

Складання лінійних програм мовою C++ , стандартні числові функції.

Тема 1.3 Розгалуження.

Структура розгалуження. Умовний оператор. Вкладена конституція. Структура вибору, оператор варіанту. Оператор безумовного переходу.

Література: [1] с25-36,

[2] с.121-164

Лабораторне заняття - 2

Складання розгалужених програм мовою C++

Тема 1.4 Адреси даних. Вказівки. Динамічна пам'ять.

Адреси даних. Визначення та використання адрес. Перепризначення адрес. Вказівки. Динамічна пам'ять. Використання динамічної пам'яті при розробці програм на мові

Література: [1] с.36-46,

[2] с.164-195

Лабораторне заняття – 3

Складання циклічних програм мовою C++.

Тема 1.5 Цикли.

Оператори циклу FOR, DO, WHILE. Структури та об'єднання, порівняння. Способи доступу до вмісту структур та об'єднань. Перетворення типів даних.

Література: [1] с.46-53,

[2] с.195-202

Лабораторне заняття – 4

Робота з одновимірними масивами використання динамічної пам'яті

Тема 1.6 Масиви.

Масиви. Ініціалізація масивів та класи пам'яті. Вказівники масивів. Операції над вказівниками. Багатомірні масиви. Ініціалізація двомірного масиву. Вказівники та багатомірні масиви.

Література: [1] с.53-81,
[2] с.202-218

Лабораторне заняття - 5

Використання масивів в мові C++

Тема 1.7 Функції. Рядки

Визначення функції. Виклик функції. Зв'язок параметрів з аргументами. Отримання результату. Рекурсивні функції в мові . Функції зі змінним числом аргументів. Параметри функції MAIN. Функції роботи з стрічками символів. Функції роботи з файлами. Обробка рядків на мові Функції роботи з рядками. Виконання операцій з рядками.

Література: [1] с.81-106,
[2] с.218-233

Лабораторне заняття - 6

Складання програм з використанням функцій на мові C++

Тема 2.9 Структури, списки, стеки, черги, об'єднання.

Структури в мові . Встановлення структурного шаблону. Визначення та ініціалізація структур. Доступ до елементів структур. Масиви структур. Вкладені структури. Вказівники та структури.

Література: [1] с.122-156,
[2] с.255-278

Лабораторне заняття - 7

Процедури та функції в мові , рекурсія

Тема 2.10 Графіка.

Програмування графіки. Текстовий режим. Графічний режим. Зображення текстів. Зображення графічних об'єктів. Використання кольорової палітри. Використання графічних функцій.

Література: [1] с.156-164,

[3] с.15-16

Лабораторне заняття -8

Використання графіки

Розділ 2. Візуальні середовища програмування Visual C++

Тема 2.1 Інтерфес середовища. Палітра компонентів. Загальна характеристика компонентів.

Інтерфейс програми. Структура проекту. Панелі інструментів. Палітри компонент. Вікно коду програми.

Основні властивості та методами класу TForm. Використання в програмах компонентів Tframe, TMainMenu, TPopupMenu, TLabel, TButton.

Практичне заняття -1

Використання в програмах компонентів Tframe, TMainMenu, TPopupMenu, TLabel, TButton.

Тема 2.2 Використання компонентів палітри Standart. палітри Addition, Win32.

Використання в програмах компонентів. TEdit, TМемо. Використання перемикачів та списків в програмах.

Література: [4] с.68-93,

Тема 2.3 Використання компонентів палітри System , Dialog

Використання в програмах компонентів TDateTimePicker, TMonthCalendar, TTreeView, TListView, THeaderControl. Використання в програмах компонентів TTimer, TPaintBox.

Використання в програмах компонентів TOpenDialog, TSaveDialog, TOpenPictureDialog, TSavePictureDialog. Використання в програмах компонентів TOpenPictureDialog, TSavePictoryDialog, TFontDialog, TColorDialog, TFindDialog, TReplaceDialog.

Література: [5] с.67-86,

Література: [5] с.156-188,

Практичне заняття -2

Використання компонентів палітри Standart. Addition, Win32. Використання компонентів палітр System Dialog.

Тема 4.7 Робота з базами даних.

Створення БД, та вивід інформації. Маніпуляція полями БД

Робота з BLOB полями. Створення фільтрів, пошуку, та залежних таблиць.

Робота з таблицями в текстовому форматі.

Література: [5] с.207-286,

Практичне заняття –3
Розробка програм з використанням ADO.

Тема 4.8 Створення довідки. Поставка додатків.

Створення файлу довідки. Компілювання файлу. Підключення довідки до програми. Компілювання програми. Виготовлення інсталяційного пакету.

Література: [6] с.37-86,

Тема 4.10 Поставка додатків в архітектуру клієнт/сервер.

Особливості структури клієнт/сервер. Серверні СУБД. Перенос і інсталювання програм у комп'ютерній мережі.

Підготовка програми до інсталяції в комп'ютерній мережі.

Література: [6] с.203-251,

Практичне заняття -4

Створення довідки для програм. Підготовка інсталяційного пакету.

компонентів												
Використання компонентів палітри Standart. Win32.	6	4	2									
Використання компонентів палітри System Dialog	6	4	2									
Робота з базами даних	12	4	2			6						
Створення довідки Поставка додатків.	4	2	2									
Поставка додатків в архітектуру клієнт/сервер.	2	2										
<i>Разом за розділом 2</i>	32	18	8			6						
Курсове проектування	20				20							
Всього	90	40	8	16	20	6						

5. Темі семінарських занять

Навчальним планом не передбачені.

6. Темі практичних занять

№ зан.	Назва теми	Кількість годин
1.	Використання в програмах компонентів Tframe, TMainMenu, TPopupMenu, TLabel, TButton.	2
2.	Використання компонентів палітри Standart. Addition, Win32. Використання компонентів палітри System Dialog	2
3.	<i>Розробка програм з використанням ADO.</i>	2
	<i>Всього</i>	6

7. Теми лабораторних занять

№ зан.	Назва теми	Кількість годин
1.	Складання лінійних програм мовою C++ , стандартні числові функції.	2
2.	Складання розгалужених програм мовою C++	2
3.	Складання циклічних програм мовою C++.	2
4.	Робота з одновимірними масивами використання динамічної пам'яті	2
5.	Використання масивів в мові C++	2
6.	Складання програм з використанням функцій на мові C++	2
7.	Процедури та функції в мові , рекурсія	2
8.	Використання графіки	2
	Всього	16

8. Самостійна робота

№ зан.	Назва теми	Кількість годин
1	Тема: 2.4 Властивості компонент ADO [1] с. 218 – 322, [2] с. 216 - 237	6
2	Курсова робота	20
	Всього	26

9. Індивідуальні завдання

Курсова робота.

Курсове проектування ставить перед собою ціль прищепити студенту навички самостійного рішення конкретних задач, зв'язаних з методами і засобами реалізації на електронно-обчислювальній техніці, на основі придбаних знань при вивченні загальнотехнічних і профільюючих дисциплін. Воно повинно сприяти закріпленню, поглибленню й узагальненню знань, отриманих студентом за час навчання. З допомогою курсової роботи студент підготовляється до виконання більш складних технічних задач — дипломного проектування, розробки повноцінних програмних засобів.

При роботі студент, відповідно до завдання на проектування, вирішує конкретні технологічні й організаційно-економічні задачі. У процесі проектування він повинний виявити уміння користуватися системою розробки програмного засобу, довідковою літературою, стандартами, табличними матеріалами, періодичною й іншою літературою.

Курсова робота є першою самостійною роботою, у якій студент повинен виявити не тільки здатність до використання матеріалів проектування, але й уміння аналізувати варіанти рішень з погляду їхньої технічної, конструктивної й економічної доцільності.

Усі рішення, прийняті студентом у процесі розробки курсової роботи, повинні бути підлеглі задачам, поставлених перед інформаційними технологіями країни.

За прийняті в роботах технічні й конструктивні рішення і вірогідність різних обчислень, моделей, відповідає автор роботи (студент).

Рекомендована структура роботи

1. Титульний аркуш
2. Завдання на курсову роботу.
3. Зміст
4. Вступ
5. Основна частина
6. Висновки
7. Список використаних джерел та програмних засобів
8. Додатки.

Опис розділів курсової роботи.

1. Титульний лист є першим листом документа. Він виконується за ДСТ 2.105-95 на аркушах формату А4 (210x297 мм). Титульний аркуш курсової роботи містить найменування навчального закладу, де виконана робота; назву відділення; назву роботи; спеціальність; прізвище, ім'я, по батькові автора; науковий ступінь, вчене звання, прізвище, ім'я, по батькові керівника; рік виконання.

2. В завданні приводиться тема та об'єм курсової роботи. Темі обираються згідно варіанту та затверджуються головою циклової комісії.

3. Зміст містить найменування і номери початкових сторінок усіх розділів, підрозділів і пунктів (якщо вони мають заголовки); зокрема: вступу, висновків до розділів, загальних висновків, додатків, списку використаної літератури та ін. Зміст виконують відповідно ДО ДЕРЖСТАНДАРТУ 2.105-95.

4. Вступ розкриває сутність і стан наукової проблеми (задачі) та її значущість, підстави і вихідні дані для розробки теми, обґрунтування необхідності проведення дослідження.

Далі подається загальна характеристика курсової роботи в рекомендованій нижче послідовності.

Актуальність теми. Шляхом критичного аналізу та порівняння з відомими розв'язаннями проблеми (наукової задачі) обґрунтовують актуальність та доцільність роботи.

Мета і задачі дослідження. Не слід формулювати мету як „Дослідження...”, „Вивчення...”, тому що ці слова вказують на засіб досягнення мети, а не на саму мету.

Наукова новизна одержаних результатів. Подають коротку анотацію наукових положень (рішень), запропонованих студентом. Необхідно показати

відмінність одержаних результатів від відомих раніше, описати ступінь новизни (вперше одержано, удосконалено, дістало подальший розвиток).

Практичне значення одержаних результатів. Треба подати відомості про практичне застосування одержаних результатів або рекомендації щодо їх використання.

5. Подальший зміст основної частини проекту повинний строго відповідати завданню на проектування і закінчуватися висновком.

Далі подається загальна характеристика курсової роботи в рекомендованій нижче послідовності.

5.1 Теоретичні відомості.

5.2 Специфікація процедур і функцій

5.3 Опис констант та змінних.

5.4 Блок схеми алгоритмів.

5.5 Опис використаних компонентів.

5.6 Лістинг програми.

5.7 Результати роботи програми.

5.1 В теоретичних відомостях студент повинен описати існуючі програмні засоби, методи реалізації, компоненти та інше з посиланням на літературні джерела з зазначенням сторінок та програмні засоби з вказанням назви самого засобу.

5.2 В специфікації процедур і функцій необхідно розглянути нові логічні структури, які були використанні при написанні курсового проекту, з чітким поясненням їх функціонування, та конкретними прикладами роботи.

5.3 В описі констант та змінних слід привести основні та додаткові змінні та константи, що використовувались під час курсового проектування.

5.4 В підрозділі слід привести блок-схеми алгоритмів, що були використані в процедурах та функціях програми.

5.5 В підрозділі в заданій формі описуються компоненти, що використовувались в усіх формах проекту.

5.6 Лістинг програми містить всі наявні конструктивні компоненти готової програми. Зазначення кожного незалежного блоку необхідно виділити до правил оформлення курсової роботи.

5.7 Результати програми повинні бути описанні з використанням прикладів, які підтверджуються самою програмою. Необхідно вказати особливості результатів виходячи з вказаних у вступі і в теоретичній частині даних.

6. У висновку підводиться підсумок проробленої роботи. У ньому повинні зазначатися: оцінка результатів роботи; висновки по проробленій роботі; пропозиції по використанню отриманих результатів; шляхи і мета подальшої роботи в цій області або обґрунтованій недоцільності її продовження.

Обсяг курсової роботи до 40 сторінок машинописного тексту.

7. Список використаних джерел та програмних засобів повинен містити перелік літератури та програмних засобів, використаних при виконанні роботи. Список оформлюється за ДСТ 7.32-91 і даними рекомендаціями.

10. Методи навчання

I. По джерелу знань:

- 1.1. Словесні – лекція;
- 1.2. Діалогічні (бесіда, диспут, семінар);
- 1.3. Практичні (вправи, практичні завдання);

II. По пізнавальній діяльності:

- 2.1. Репродуктивний (відтворювальний) – розв'язання задач;
- 2.2. Частково-пошуковий (самостійна робота),
- 2.3. Евристичний (проблемно-розвиваючий) – евристична бесіда, евристичне фронтальне опитування;
- 2.4. Метод проблемного викладання;

III. Методи по логіці навчання:

- 1.1. Індуктивний – від часткового до загального;
- 1.2. Дедуктивний – від загального до часткового;
- 1.3. Синтез;
- 1.4. Аналіз;
- 1.5. Порівняння.

IV. Методи колективної розумової діяльності:

- 4.1. Пізнавальна суперечка;
- 4.2. Навчальні дискусії;
- 4.3. Метод „мозкового штурму”;

V. Імітаційні методи навчання (ділова гра).

11. Методи контролю

Види контролю	Застосування
Поточний контроль	Атестація, проведення семінарських занять, усний контроль знань, модульний контроль знань, контроль виконання самостійної роботи студентів
Підсумковий контроль	Семестровий контроль (екзамен)

12. Критерії оцінювання

Підсумковий контроль знань студентів з дисципліни «Програмування» проводиться у формі екзамену.

Згідно з вимогами регламенту навчального процесу в коледжі, викладачі зобов'язані контролювати відвідування студентами лекцій, семінарських та лабораторних занять, здійснювати систематичний поточний контроль знань студентів протягом всього строку вивчення дисципліни. За результатами такого контролю студенти допускаються або не допускаються до складання екзамену.

Допуск до екзамену студент отримує від викладача, який протягом семестру вів у групі лекції, семінарські і лабораторні заняття та курсова робота з

цієї дисципліни. Студенти, які мали заборгованість (пропуски занять, незадовільні оцінки, невиконана курсова робота) і не ліквідували її до початку сесії, до заліку не допускаються.

1. "Відмінно" - студент дає обґрунтовані, глибокі й теоретично правильні відповіді на поставлені питання; правильно і раціонально виконує практичні завдання, обґрунтовуючи їх вичерпними поясненнями; робить висновки та узагальнення; використовує сучасні прилади для виконання поставленого завдання; вміло виконує лабораторні роботи, виконує складання програм будь якого рівня, не допускаючи помилок, створює програмні засоби, використовуючи оптимальні алгоритми.

2. "Добре" - студент володіє знаннями матеріалу на рівні вимог, наведених вище, але у розкритті питань ним були допущені незначні помилки у термінах; незначні неточності у виконаних практично завданнях, що не суттєво впливають на розрахунки і роботу програмних засобів.

3. "Задовільно" - студент допускає значні помилки при відповіді на теоретичні запитання: при виконанні практичних завдань необхідні незначні підказки з боку вчителя, але отримані програмні продукти являються працездатними і виконують базові функції;

4. "Незадовільно" - студент допускає неправильні відповіді або зовсім не відповідає на теоретичні запитання; не виконує практичних завдань; не вміє підібрати інструмент рішення поставлених завдань.

13. Методичне забезпечення

1. Методичні вказівки до самостійної роботи студентів з предмету «Програмування»;
2. Методичні матеріали до поточного і підсумкового контролю:
 - завдання до проведення тематичного контролю знань;
 - завдання до проведення контрольних робіт;
 - завдання та методичні вказівки до курсового проектування;
 - питання до екзамену;

14. Рекомендована література

Основна

1. Коноваленко І.В, Марущак П.О. Платформа .NET та мова програмування C# 8.0. ТНТУ, 2020 332 с
2. Коноваленко І.В. Програмування мовою C# 6.0. ТНТУ, 2016. 227 с.
3. C# Концепція та синтакси Б.М. Голуб, Львів 2018р. 136с.
4. Розробка додатків засобами мови програмування C#: Навч.-метод. посібник для проведення лабораторних робіт для студентів вищих навчальних закладів спеціальності «системний аналіз» /Є.В.Івохін, М.Ф.Махно, О.Г.Піскунов. – К.: Видавничо-поліграфічний центр "Київський університет", 2021. – 100с.

5. О.С. Зеленський_С Посібник для 10-11 класів, 2020р 189с.
6. Єжова Л.Ф. Алгоритмізація і програмування процедур обробки інформації: Навч.-метод. посібник. – К.: КНЕУ, 2019р. 60с.
7. Booch В. Object-Oriented Analysis and Design with Applications. 3rd edition. Addison-Wesley. 720 p.

Додаткова

8. Perkins В. Beginning C# 7 Programming with Visual Studio 2017 / Benjamin Perkins, Jacob Vibe Hammer, Jon D. Reid. – Indianapolis, IN: Wrox, 2018. – 884 с.
9. Халецька З.П. Математична логіка та теорія алгоритмів: Навчальний посібник / З.П. Халецька, В.В. Наратовий. – Кропивницький: РВВ КДПУ ім. В. Винниченка, 2017. – 128 с.
10. Завада О. П. Алгоритмізація і програмування: Тексти лекцій. – Львів: Видавничий центр ЛНУ імені Івана Франка, 2004. - 76 с.
11. Алгоритмізація і програмування: Навчальна практика 2022р, 35с.

Інформаційні ресурси

1. C# Підручник. Режим доступу: <https://w3schoolsua.github.io/cs/index.html#gsc.tab=0>
2. Документація по Visual Studio – Режим доступу: <https://docs.microsoft.com/visualstudio/ide/?view=vs-2022>
3. C# Programming. Yellow Book [Ел. ресурс]. URL: <https://www.robmiles.com/cellow-book>
4. 2. C# Tutorial and source code. – Режим доступу: <http://csharp.net-informations.com>
5. 3. The Visual Studio Blog. – Режим доступу: <https://blogs.msdn.microsoft.com/visualstudio>
6. 4. .NET Tutorial - Hello World in 10 minutes. – Режим доступу: <https://www.microsoft.com/net/learn/get-started-with-dotnet-tutorial>
7. 5. Microsoft Docs. – Режим доступу: <https://docs.microsoft.com/ru-ru/?view=vs-2017>