

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
КИЇВСЬКИЙ АвіАЦІЙНИЙ ФАХОВИЙ КОЛЕДЖ

Найменування центрального органу управління освітою, власника

«ЗАТВЕРДЖУЮ»

Заст. директора



Алла АДАМЕНКО

«26» серпня 2022 р.

ОБ'ЄКТНО-ОРІЄНТОВАНЕ ПРОГРАМУВАННЯ

Назва навчальної дисципліни

ПРОГРАМА

навчальної дисципліни

підготовки фахового молодшого бакалавра


Назва освітньо-професійного ступеня

спеціальності 121 Інженерія програмного забезпечення

Шифр і назва спеціальності

РОЗРОБЛЕНО ТА ВНЕСЕНО: Київським авіаційним фаховим коледжем

Повне найменування вищого навчального закладу

Розробники:  Т.Обгарук — Викладач
Підпис вищої категорії

Схвалено на засіданні циклової комісії
професійної та практичної підготовки

Назва циклової комісії

Протокол № 1, від «25» 08 2022р.

Голова циклової комісії


Підпис

А. Сухомутька
Прізвище та ініціали

ВСТУП

Програма вивчення навчальної дисципліни «Об'єктно-орієнтоване програмування» складена відповідно до освітньо-професійної програми підготовки фахового молодшого бакалавра спеціальності 121 «Інженерія програмного забезпечення».

Предметом вивчення дисципліни «Об'єктно-орієнтованого програмування» є фундаментальні поняття об'єктно-орієнтованого програмування, основні складові цієї парадигми, а також основні методи та прийоми об'єктно-орієнтованого програмування, які включають аналіз, проектування, еволюцію та модифікацію об'єктів.

Міждисциплінарні зв'язки: Навчальна дисципліна «Об'єктно-орієнтоване програмування» входить до циклу професійних дисциплін, що формують фахівців у галузі комп'ютерних наук.

Викладання даної дисципліни базується на курсах:

- вступ до спеціальності.
- основи програмування та алгоритмічні мови;
- системне програмування та операційні системи.

Програма навчальної дисципліни складається з таких тем:

- Тема 1. Основні оператори C++.
- Тема 2. Масиви.
- Тема 3. Основні парадигми в програмуванні.
- Тема 4. Класи, об'єкти.
- Тема 5. Мова моделювання UML.
- Тема 6. Бібліотеки мови C++
- Тема 7. Спадкування.
- Тема 8. Динамічні структури даних
- Тема 9. Поліморфізм.
- Тема 10. Бібліотека шаблонних класів STL

1. Мета та завдання навчальної дисципліни

1.1 Метою викладання навчальної дисципліни «Об'єктно-орієнтоване програмування» є одержати знання про сучасний стан та тенденції ООП, про важливі складові цієї парадигми, такі як шаблони класів, поліморфізм, візуальне моделювання об'єктів та їх відношень, вчити студентів основним методам та прийомам об'єктно-орієнтованого програмування, яке включає аналіз, проектування, еволюцію та модифікацію об'єктів; ознайомити з особливостями єдиної термінології ООП, прищепити їм вміння ефективно використовувати ООП у програмуванні реальних процесів при розробці інформаційних систем, а також прикладних програм що застосовуються в різних галузях.

1.2.Завдання – формування знань про принципи об'єктно-орієнтованого програмування, про конкретні методи та прийоми програмування на об'єктній мові C++, та в об'єктно-орієнтованому візуальному середовище програмування, а також, вивчення та засвоєння загальних принципів побудови систем, які основані на сучасних технологіях об'єктно-орієнтованого програмування.

1.3. Згідно з вимогами освітньо-професійної програми студенти повинні:

знати:

- основні поняття і терміни об'єктно-орієнтованого програмування;
- принципи та методи побудови об'єктно-орієнтованих програм мовами C++ ;
- принципи та методи побудови об'єктно-орієнтованих програм у візуальному середовище програмування ;
- методологію об'єктного програмування та принципи застосування об'єктної декомпозиції;

вміти:

- здійснювати декомпозицію об'єктів,
- візуально описувати модель їхньої взаємодії в об'єктно-орієнтованій інформаційній системі;
- використовувати переваги ООП при побудові складних програм, що включають багаторівневу ієрархію об'єктів.

На вивчення навчальної дисципліни відводиться **216** годин за навчальним планом.

2. Інформаційний обсяг навчальної дисципліни

Тема 1. Основні оператори C++

Типи даних, модифікатори. Вирази, перетворення типів, приведення типів. Арифметичні операції, операції інкрементації та декрементації, операції порівняння, логічні операції, побітові операції. Вбудовані математичні функції. Вирази. Перетворення типів. Основні функції роботи з рядками. Локальні та глобальні змінні.

Оператори присвоєння, складений та множинний оператори присвоєння, тернарний оператор, різновиди умовного оператора, оператори циклів, оператор вибору. Структури. Функції.

Тема 2. Масиви

Одновимірні та двовірні масиви. Типові операції з масивами. Опис змінних, масивів. Одновимірні та багатовимірні масиви. Масиви рядків. Вкладений оператор циклу. Застосування операторів циклу. Приклади. Масиви структур. Передача масивів у функцію. Передача структур у функцію.

Тема 3. Основні парадигми в програмуванні

Класифікація мов програмування. Основні напрямки в програмуванні. Основні принципи об'єктно-орієнтованого програмування. Роль предмету. Переваги і цілі об'єктно-орієнтованого програмування. Принцип абстрагування. Принцип обмеження доступу. Модульність. Етапи розробки програмних систем із використанням об'єктно-орієнтованого програмування: аналіз, проектування.

Тема 4. Класи, об'єкти

Клас як тип даних. Рівні доступу до елементів класу: відкритий, захищений, приватний. Методи класу. Змінні типу класу. Інтерфейс класу. Реалізація класу. Приклади інтерфейсу і реалізації. Основні характеристики об'єкту. Об'єктно-орієнтований підхід у програмуванні. Створення класу. Об'єкти, функції-члени, дані-члени. Конструктори, деструктори. Стан та поводження об'єктів. Дружні функції. Масиви об'єктів. Ініціалізація об'єктів.

Тема 5. Мова моделювання UML.

Основні поняття мови моделювання UML. Нотації мови UML. Види сутностей в діаграмі класів: структурні, поведінкові, анотуючі. Типи діаграм, діаграми класів, діаграма прецедентів, діаграма послідовностей. Основні компоненти мови: діаграми, сутності, зв'язки. Типи діаграм в мові UML, діаграма класів. Правила оформлення діаграми класів, варіанти запису кратності атрибутів. Типи відношень між класами. Приклади.

Тема 6. Бібліотеки мови C++

Бібліотеки стандартних функцій: функції введення-виведення, функції для роботи з рядками та символами, бібліотека математичних функцій, функції часу і дати, функції динамічного розподілу пам'яті, службові функції.

Бібліотеки стандартних класів: стандартні класи введення-виведення, стандартні контейнерні класи, стандартні алгоритми.

Тема 7. Спадкування

Базові та похідні класи. Види спадкувань. Батьківський і похідний класи. Визначення похідного класу. Конструктори та деструктори похідних класів. Перевизначення методів і властивостей. Відкрите та закрите спадкування. Захищені члени класу, захищене спадкування.

Тема 8. Динамічні структури даних

Загальна характеристика динамічних структур даних. Визначення. Лінійний однозв'язний та двозв'язні списки, циклічний список, черга, стек.

Створення лінійного однозв'язного списку, операції з лінійним однозв'язним списком. Застосування покажчика для створення і виконання операцій з лінійним однозв'язним списком. Доступ до елементів списку, пошук елементів в лінійному однозв'язному списку. Приклади програм. Різновиди лінійних списків. Організація лінійних списків за допомогою покажчиків.

Тема 9. Поліморфізм

Поняття поліморфізму. Перевантаження функцій. Віртуальні функції. Абстрактні класи. Перевизначення та перевантаження як типи поліморфізму. Ефективність поліморфізму. Перевантаження конструкторів. Ініціалізовані та неініціалізовані об'єкти.

Створення операторної функції-члена. Перевантаження операторів додавання та віднімання. Створення префіксної та постфіксної форм операторів інкрементації та декрементації.

Перевантаження операторів за допомогою дружньої операторної функції. Застосування дружньої функції для перевантаження операторів «++», «--». Підвищення гнучкості програми за допомогою дружньої операторної функції.

Тема 10. Бібліотека шаблонних класів STL

Склад бібліотеки. Контейнерні класи, алгоритми, ітератори, різновиди ітераторів, предикати, різновиди предикатів, функції порівняння, функтори, розподілювачі. Призначення алгоритмів. Стандартні алгоритми. Застосування алгоритмів для роботи з контейнерними класами. Приклади. Клас Vector. Функції-члени класу. Функції-члени класу Vector та алгоритми для роботи з класом Vector. Клас Stack. Функції-члени класу.

3. Рекомендована література

1. Дудзяний І.М. Об'єктно-орієнтоване моделювання програмних систем. Навч. Посібник. – Львів: Видавничий центр ЛНУ імені Івана Франка, 2017. –108с.
2. UML: історія, специфікація, бібліографія [Електронний ресурс] – Режим доступу: // <http://it.ridne.net/node/265>
3. Мова UML Resource Page [Електронний ресурс] –Режим доступу: // <http://www.uml.org/>
4. Unified Modeling Language – Вікіпедія [Електронний ресурс] – Режим доступу: // http://uk.wikipedia.org/wiki/Unified_Modeling_Language
5. UML: історія, специфікація, бібліографія [Електронний ресурс] – Режим доступу: // <http://it.ridne.net/node/265>
6. UML Resource Page [Електронний ресурс] – Режим доступу: // <http://www.uml.org/>
7. Бублик В.В. Об'єктно-орієнтоване програмування–К.: ІТ-книга, 2015. –640с. [Електронний ресурс] – Режим доступу: http://itknyga.com.ua/docs/OOP_final.pdf
8. Інформатика та обчислювальна техніка : Короткий тлумачний словник / В. П. Гондюл, А. Г. Дерев'яно, В. В. Матвеев, Ю. З. Прохур; за ред. проф. В. П. Гондюла. – К. : Либідь, 2000. – 320 с
9. С++. Основи програмування. Теорія та практика : підручник / [О.Г. Трофименко, Ю.В. Прокоп, І.Г. Швайко, Л.М. Буката та ін.] ; за ред.О.Г.Трофименко. – Одеса: Фенікс, 2010. – 544 с

4. Форма підсумкового контролю успішності навчання: залік, екзамен.

5.Засоби діагностики успішності навчання: Комплексний державний екзамен.