Міністерство освіти і науки України

**Ржищівський**

**індустріально-педагогічний технікум**

 **„**ЗАТВЕРДЖУЮ**“**

 *заступник директора по НВР*

 Л. Г. Цюцюра

 „ “ 2017року

**НАВЧАЛЬНА ПРОГРАМА**

з дисципліни

**„Об’єктно-орієнтоване програмування“**

для студентів денної форми навчання

напрям підготовки 0501 „Інформатика та обчислювальна техніка“

(12 „Інформаційні технології“)

за спеціальністю 5.05010301 „Розробка програмного забезпечення“

(121 „Інженерія програмного забезпечення“)

2017

Навчальна програма дисципліни „Об’єктно-орієнтоване програмування“

для студентів денної форми навчання

за спеціальністю 5.05010301 „Розробка програмного забезпечення“ (121 „Інженерія програмного забезпечення“)

Ржищівського індустріально-педагогічного технікуму – Ржищів, 2017. –

Укладачі: Бабич Тетяна Миколаївна

Рецензенти:

Ільїна Світлана Василівна, Бабич Олександр Семенович

Навчальну програму обговорено та схвалено на засіданні Педагогічної ради

Від „30“ серпня 2017 р., протокол №1

Рецензія

на навчальну програму
з дисципліни
„Об’єктно-орієнтоване програмування“

зі спеціальності 5.05010301 „Розробка програмного забезпечення“
(121 „Інженерія програмного забезпечення“)

Подана до розгляду навчальна програма з дисципліни „Об’єктно-орієнтоване програмування“ розроблена з урахуванням вимог ОПП та навчальних планів для навчальних закладів, що здійснюють підготовку молодших спеціалістів за спеціальністю 5.05010301 „Розробка програмного забезпечення“ (121 „Інженерія програмного забезпечення“).

В програмі передбачається викладення навчального матеріалу за 15 темами у логічній послідовності з урахуванням сучасних вимог до рівня знань молодшого спеціаліста. Загальний обсяг дисципліни за програмою складає 162 годин, з яких 136 годин аудиторних, що становить 83% від загального обсягу і 26 годин для самостійної роботи студентів, що становить 17% від загального обсягу.

У пояснювальній записці сформульовано мету й завдання дисципліни „Об’єктно-орієнтоване програмування“. Програму структуровано: визначено зміст навчання та основні вимоги до рівня знань, умінь і навичок студентів. У програмі визначено чіткі вимоги до знань і вмінь студентів, що сприяє як практичному спрямуванню викладання об’єктно-орієнтованого програмування, так і міцному засвоєнню теоретичного матеріалу. Вивчення програмового матеріалу побудоване з урахуванням міжпредметних зв’язків.

Викладання дисципліни за поданою програмою дозволить отримати знання з теорії об’єктно-орієнтованого програмування.

На основі викладеного матеріалу вважається доцільним схвалити і рекомендувати подану програму для викладання дисципліни „Об’єктно-орієнтоване програмування“ у Ржищівському індустріально-педагогічному технікумі.

Рецензент:

**ПОЯСНЮВАЛЬНА ЗАПИСКА**

Програма предмету “Об’єктно-орієнтоване програмування” передбачає вивчення Microsoft Visual Studio як систему об’єктно-орієнтованого програмування. В програмі детально описані питання роботи в інтегрованому середовищі розробки C#, зроблено огляд компонентів бібліотеки, а також розглянуті питання побудови графічних зображень. Крім того особлива увага приділяється практикуму з програмування, створенню ігрових додатків.

Модель об’єктно-орієнтованого програмування ґрунтується на трьох основних концепціях: інкапсуляція, спадкування, поліморфізм. Інкапсуляція – це механізм, який об’єднує дані і методи, які використовуються для обробки цих даних. При цьому механізм інкапсуляції захищає дані і методи від стороннього втручання. Спадкування – це процес, завдяки якому один об’єкт може набувати властивості іншого. Це означає, в ООП на основі уже існуючих класів можна будувати похідні класи. Поліморфізм – це властивість коду вести себе по різному в залежності від ситуації, що виникає в момент виконання програми.

Завдання викладача – дати студентам знання сучасних інформаційних технологій, навчити створювати прикладні програми. Навчити опрацьовувати науково-технічну літературу, періодичні видання з метою оволодіння новими знаннями.

В результаті вивчення предмету студенти повинні мати чітку уяву про такі поняття: клас, об’єкт, властивості, методи, події, принцип інкапсуляції, принцип поліморфізму, принцип спадкування.

Студент повинен знати:

* призначення об’єктно-орієнтованого програмування;
* основні концепції об’єктно-орієнтованого програмування: інкапсуляція, спадкування, поліморфізм;
* типи даних в мові С#;
* складені типи даних: масиви, рядки, структури, перерахування, класи;
* компоненти бібліотеки C#;
* графіку, графічні примітиви, бітові образи.

Студент повинен уміти:

* здійснювати постановку задачі, включати об’єкти в свою програму (візуальне програмування); створювати прикладні програми;
* формалізацію і алгоритмізацію задачі;
* правильно ініціалізувати змінні;
* застосовувати структури слідування, розгалуження, циклічні структури;
* створювати проекти на мові С#.

Предмет вивчається на другому курсі у четвертому семестрі і продовжується його вивчення на третьому курсі у п’ятому семестрі. Курс присвячено вивченню компонентів C# та створенню додатків (практикум по програмуванню). Значну увагу слід приділяти самостійному опрацюванню матеріалу та творчій роботі студентів.

Екзамен студенти складають наприкінці п’ятого семестру. Екзаменаційні білети попередньо надаються. Завершується вивчення курсу з „Об’єктно-орієнтованого програмування” написанням курсової роботи.

**Тематичний план
з дисципліни
„Об’єктно-орієнтоване програмування”.**

|  |  |  |
| --- | --- | --- |
| №п/п | Назва теми | Обсяг годин за навчальною програмою |
| всього | аудит. | лекції | лабор-практ. | самост. | контр. роботи |
| 1 | Вступ. Основи CLR. Модель виконання коду в середовищі CLR.Етапи розв’язку задач. організація робочого простору в комп’ютері. Елементи мови C#. | 10 | 8 | 4 | 4 | 2 |  |
| 2 | Змінні. Базові типи змінних. Концепція пам’яті. Арифметичні операції. Приорітети операцій. Зведення типів. Логічні операції. | 10 | 8 | 4 | 4 | 2 |  |
| 3 | Керуючі структури мови С#.Структура вибору if/else. Тернарна структура вибору. Перемикач switch. | 12 | 10 | 6 | 4 | 2 |  |
| 4 | Циклічні структури мови С#.Структура повторення for. Структура повторення while. Структура повторення do/while. Оператори переривання break та continue. | 14 | 12 | 6 | 6 | 2 |  |
| 5 | Область дії змінних. | 8 | 6 | 4 | 2 | 2 |  |
| 6 | Форматування даних. Помилки часу виконання. Способи їх попередження. Метод TryParse(). Оператор try{} catch {}. Властивість e.Handled. | 8 | 6 | 4 | 2 | 2 |  |
| 7 | Масиви. Ініціалізація елементів масиву. Типові задачі. Базовий клас System.Array. Сортування елементів масиву. Пошук заданого елементу в масиві. Перестановка елементів масиву. Визначення максимального і мінімального елементу масиву. Генератор випадкових чисел. | 20 | 18 | 6 | 12 | 2 |  |
| 8 | Класи і об’єкти. Типи функцій. Доступ до елементів класу. Конструктор. Деструктор. Виділення та вивільнення пам’яті. Автоматичне управління пам’яттю (збір сміття). Робота на платформі, яка підтримує збір сміття. Алгоритм збору сміття. Вивільнення ресурсів за допомогою механізму фіналізації. | 14 | 12 | 6 | 6 | 2 |  |
| 9 | Статичні елементи класу. Програма – клас на мові С#. Перевантаження функцій. Аргументи функцій. Компоновка проекту. | 8 | 8 | 4 | 4 | 0 |  |
| 10 | Символи, рядки і опрацювання тексту. Символи. Тип System.String. Рядки. Перетворення числових рядків. | 14 | 12 | 6 | 6 | 2 |  |
| 11 | Основні концепції об’єктно-орієнтованого програмування. Наслідування. Інкапсуляція. Поліморфізм | 8 | 6 | 4 | 2 | 2 |  |
| 12 | Ієрархія класів. Потоки. Консольне введення/виведення. Файлове введення/виведення. | 8 | 6 | 4 | 2 | 2 |  |
| 13 | Константи і поля. Методи. Параметри. Властивості. Події. | 10 | 8 | 4 | 4 | 2 |  |
| 14 | Дії над об’єктами. Оголошення і взаємодія об’єктів. Об’єкти і функції. Віртуальні функції. Перевантаження операторів. | 6 | 6 | 4 | 2 | 0 |  |
| 15 | Проектування типів. Всі типи – похідні від System.Object. Простір імен. | 12 | 10 | 6 | 4 | 2 |  |
|  | ***Всього*** | ***162*** | ***136*** | ***72*** | ***64*** | ***26*** |  |

**Програма дисципліни „Об’єктно-орієнтоване програмування”**

*1. Платформа .NET. Система Common Language Runtime (CLR).
Модель виконання коду в середовищі CLR. Мова С#.*

Вступ. Основи CLR. Модель виконання коду в середовищі CLR.

Етапи розв’язку задач. Організація робочого простору в комп’ютері. Елементи мови C#.

Змінні. Базові типи змінних. Концепція пам’яті. Арифметичні операції. Приорітети операцій. Зведення типів. Логічні операції.

Керуючі структури мови С#. Структура вибору if/else. Тернарна структура вибору. Перемикач switch. Практикум з програмування.

Циклічні структури мови С#. Структура повторення for. Структура повторення while. Структура повторення do/while. Оператори переривання break та continue. Практикум з програмування.

Область дії змінних. Форматування.

*2. Масиви у мові С#.*

Масиви. Ініціалізація елементів масиву. Типові задачі. Базовий клас System.Array. Сортування елементів масиву. Пошук заданого елементу в масиві. Перестановка елементів масиву. Визначення максимального і мінімального елементу масиву. Генератор випадкових чисел.

Символи, рядки і опрацювання тексту. Символи. Тип System.String. Рядки. Перетворення числових рядків.

*3. Основні концепції об’єктно-орієнтованого програмування.*

Основні концепції об’єктно-орієнтованого програмування. Наслідування. Інкапсуляція. Поліморфізм. Класи і об’єкти. Типи функцій. Доступ до елементів класу. Конструктор. Деструктор. Виділення та вивільнення пам’яті. Автоматичне управління пам’яттю (збір сміття). Робота на платформі, яка підтримує збір сміття. Алгоритм збору сміття. Вивільнення ресурсів за допомогою механізму фіналізації.

Статичні елементи класу. Програма – клас на мові С#. Перевантаження функцій. Аргументи функцій. Компоновка проекту. Ієрархія класів. Потоки. Консольне введення/виведення. Файлове введення/виведення.

Константи і поля. Методи. Параметри. Властивості. Події.

Дії над об’єктами. Оголошення і взаємодія об’єктів. Об’єкти і функції. Віртуальні функції. Перевантаження операторів. Проектування типів. Всі типи – похідні від System.Object. Простір імен.

**Тематика та зміст самостійної роботи студентів.**

|  |  |  |
| --- | --- | --- |
| № теми | *Зміст роботи* | К-ть годин. |
| 1 | Вступ. Основи CLR. Модель виконання коду в середовищі CLR. Етапи розв’язку задач. організація робочого простору в комп’ютері. Елементи мови C#. | 2 |
| 2 | Змінні. Базові типи змінних. Концепція пам’яті. Арифметичні операції. Приорітети операцій. Зведення типів. Логічні операції. | 2 |
| 3 | Керуючі структури мови С#. Структура вибору if/else. Тернарна структура вибору. Перемикач switch. | 2 |
| 4 | Циклічні структури мови С#. Структура повторення for. Структура повторення while. Структура повторення do/while. Оператори переривання break та continue. | 2 |
| 5 | Область дії змінних. | 2 |
| 6 | Форматування. | 2 |
| 7 | Масиви. Ініціалізація елементів масиву. Типові задачі. Базовий клас System.Array. Сортування елементів масиву. Пошук заданого елементу в масиві. Перестановка елементів масиву. Визначення максимального і мінімального елементу масиву. Генератор випадкових чисел. | 2 |
| 8 | Класи і об’єкти. Типи функцій. Доступ до елементів класу. Конструктор. Деструктор. Виділення та вивільнення пам’яті. Автоматичне управління пам’яттю (збір сміття). Робота на платформі, яка підтримує збір сміття. Алгоритм збору сміття. Вивільнення ресурсів за допомогою механізму фіналізації. | 2 |
| 9 | Символи, рядки і опрацювання тексту. Символи. Тип System.String. Рядки. Перетворення числових рядків. | 2 |
| 10 | Основні концепції об’єктно-орієнтованого програмування. Наслідування. Інкапсуляція. Поліморфізм | 2 |
| 11 | Ієрархія класів. Потоки. Консольне введення/виведення. Файлове введення/виведення. | 2 |
| 12 | Константи і поля. Методи. Параметри. Властивості. Події. | 2 |
| 13 | Дії над об’єктами. Оголошення і взаємодія об’єктів. Об’єкти і функції. Віртуальні функції. Перевантаження операторів. | 1 |
| 14 | Проектування типів. Всі типи – похідні від System.Object. Простір імен. | 1 |
| **Всього:** | 26 |

Л**ітература**

Базова:

1. Троелсен Эндрю Язык программирования С# и платформа .NET: Пер. с англ. – М. : ООО „И. Д. Вильямс“, 2013. – 1392 с.: ил. – Парал. тит. англ. ISBN 978 – 5 – 8459 – 1682 – 2.
2. Рихтер Дж. С#. Программирование на платформе Microsoft .NET Framework 4.0 на языке С#. 3-е изд. - СПб.: Питер, 2012.-928 с.: ил.
3. А. Я. Архангельский Программирование в С++ Builder 6. – М.: ЗАО «Издательство БИНОМ», 2004 г. – 1152 с.: ил.
4. Вирт Н. Алгоритмы и структуры данных / Пер. с англ. – М.: Мир, 1989. – 360 с., ил.
5. Ковалюк Т. В. Основи програмування. – К.: Видавнича група BHV, 2005/ - 384 c.: іл.

Допоміжна:

1. Гринзоу Лу. Философия программирования для Windows 95/NT / Пер. с англ.- СПб.: Символ-Плюс, 1997. – 640 с., ил.
2. Джейс Либерти Освой самостоятельно С++ за 21 день: 3-е изд. пер. с англ.: Уч. пос. – М.: Издательский дом „Вильямс”, 2001. – 816 с.: ил.
3. Зелковиц М., Шоу А., Геннон Дж. Принципы разработки программного обеспечения / Пер. с англ. –М.: Мир, 1982. – 386 с., ил.
4. Культин Н. Б. С / С# в примерах и задачах СПб. БХВ – Петербург, 2001. – 228 с., ил.
5. Практическое руководство по программированию / Пер. с англ. Б. Мик, П.Хит, Н. Рашби и др.; под ред. Б. Мика, П. Хит, Н. Рашби. – М.: Радио и связь, 1986. – 168 с., ил.
6. Прата Стивен Язык программирования С++. Лекции и упражнения. Учебник: Пер. с англ./ Стивен Прата – СПб.:ООО «ДиаСофтЮП», 2003.-1104 с.
7. Страуструп Б. Язык программирования С++ М.: Радио и связь, 1991.
8. Фокс Дж. Программное обеспечение и его разработка / Пер. с англ. – М.: Мир, 1985. – 386 с., ил.
9. Харви Дейтел, Пол Дейтел Как программировать на С++: Третье издание. Пер. с англ. – М.: ЗАО «Издательство БИНОМ», 2001 г. – 1152 с.: ил.