Міністерство освіти і науки України

Ржищівський індустріально-педагогічний технікум

Циклова комісія „Інформаційних технологій“

 ЗАТВЕРДЖУЮ

 Заступник директора з НВР

 Л. Г. Цюцюра

 “ ” 2017 року

Робоча програма навчальної дисципліни

**„Основи програмування та алгоритмічні мови“**

підготовки молодшого спеціаліста

напряму підготовки: 0501 „Інформатика та обчислювальна техніка“

(12 „Інформаційні технології“)

спеціальності: 5.05010301 „Розробка програмного забезпечення“

(121 „Інженерія програмного забезпечення“)

2017

Робоча програма, методичні вказівки та індивідуальні завдання до вивчення дисципліни „Основи програмування та алгоритмічні мови“ для студентів, які навчаються за напрямом підготовки 0501 „Інформатика та обчислювальна техніка“ (12 „Інформаційні технології“), спеціальністю 5.05010301 „Розробка програмного забезпечення“ (121 „Інженерія програмного забезпечення“).

Наведені загальні методичні вказівки до вивчення дисципліни „Основи програмування та алгоритмічні мови“, література, що рекомендується, робоча програма та пояснення до окремих тем, перелік практичних занять.

Призначена для студентів, які навчаються за напрямом підготовки 0501 „Інформатика та обчислювальна техніка“ (12 „Інформаційні технології“), спеціальністю 5.05010301 „Розробка програмного забезпечення“ (121 „Інженерія програмного забезпечення“) денної форми навчання.

Укладачі Т. М. Бабич.

Робочу програму схвалено на засіданні відділення (циклової комісії) „Інформаційних технологій“.

Протокол від „30“серпня 2017 року № 1

**АНОТАЦІЯ**

Навчальна дисципліна “Основи програмування та алгоритмічні мови” передбачає вивчення основних питань з програмування, взявши за основу робочою мову програмування Microsoft Visual C++.

Метою вивчення дисципліни “Основи програмування та алгоритмічні мови” є надання знань по теоретичних основах алгоритмізації: поняття та властивості алгоритмів, базові алгоритмічні структури, типи алгоритмів, структури даних. В курсі дисципліни “Основи програмування та алгоритмічні мови” передбачено розв’язання великої кількості різноманітних задач з обробки масивів, рядків, структур. Подані програмні реалізації типових класичних алгоритмів сортування та пошуку.

В результаті вивчення предмету студенти повинні мати чітку уяву про поняття алгоритму розв’язку задачі, керуючі структури (структура слідування, розгалуження, циклічна структура).

Студент повинен знати:

* 1. призначення мов програмування високого рівня, історію їх розвитку, призначення машинної мови та асемблера;
	2. призначення транслятора (компілятор або інтерпретатор);
	3. прості типи даних: bool, unsigned short int, short int, unsigned long int, long int, int, char, float, double;
	4. складені типи даних: масиви, рядки, структури, перерахування, покажчики, класи;
	5. структуру слідування, структуру розгалуження, циклічну структуру;
	6. основні символи схем алгоритмів;
	7. функції, прототип функції, виклик функції, опис функції.

Студент повинен уміти:

1. здійснювати постановку, формалізацію і алгоритмізацію задачі;
2. правильно ініціалізувати змінні;
3. застосовувати структури слідування, розгалуження, циклічні структури;
4. писати програми на C++.

Місце дисципліни в навчальному процесі: навчальна дисципліна “Основи програмування та алгоритмічні мови” входить до циклу природничо-наукових дисциплін, які формують фахівців з програмування.

Вивчення дисципліни “Основи програмування та алгоритмічні мови” базується на знаннях математики, математичної логіки, вищої математики, інформатики, архітектури комп’ютерів, викладення яких іде поряд з вивченням основ програмування та алгоритмічних мов.

В свою чергу знання з основ програмування та алгоритмічних мов є передумовою вивчення об’єктно-орієнтованого програмування, інструментальних засобів візуального програмування, технології програмування та створення програмних продуктів.

Навчальним планом передбачається вивчення дисципліни у вигляді лекційних та практичних занять і самостійної роботи студентів (СРС).

Предмет вивчається на другому курсі у третьому та четвертому семестрах, а також на третьому курсі у п’ятому семестрі.

Контроль за вивченням курсу передбачається у вигляді складання екзамену. Екзамен студенти складають наприкінці п’ятого семестру.

Програму складено у відповідності до вимог сформульованих в ОКХ і ОПП “Стандарту вищої освіти” на підготовку молодших спеціалістів кваліфікації технік-програміст.

**Опис навчального плану з дисципліни**

**“Основи програмування та алгоритмічні мови”**

для студентів, які навчаються

за напрямом підготовки 12 „Інформаційні технології“

за спеціальністю 121 „Інженерія програмного забезпечення“

* + 1. **Опис навчальної дисципліни**

|  |  |  |
| --- | --- | --- |
| Найменування показників | Галузь знань, напрям підготовки, освітньо-кваліфікаційний рівень | Характеристика навчальної дисципліни |
| денна форма навчання | заочна форма навчання |
| Кількість кредитів національних – 3ECTS – 4,5 | Галузь знань12 „Інформаційні технології“ | Дисципліна циклу природничо-наукової підготовки |
| Напрям підготовки12 „Інформаційні технології“ |
| Загальна кількістьгодин – 162 | Спеціальність:121 „Інженерія програмного забезпечення“ | Рік підготовки |
| 2, 3 -й |  |
| Семестр |
| 3, 4, 5-й |  |
| Лекції |
| Тижневих годин для денної форми навчання:аудиторних - 124 самостійної роботи студента – 38. | Освітньо-кваліфікаційний рівень:молодший спеціаліст | 70 - год. |  |
| Практичні, семінарські |
|  |  |
| Лабораторно-практичні |
| 54 - год. |  |
| Самостійна робота |
| 38 - год. |  |
| Індивідуальні завдання: |
| 2 год |
| Вид контролю: |
| екзамен |  |

Примітка.

Співвідношення кількості годин аудиторних занять до самостійної (%):

для денної форми навчання –76 % (24 %)

**Розподіл навчального часу по курсах та семестрах**

**навчальна дисципліна**

**“Основи програмування та алгоритмічні мови”**

|  |  |  |  |
| --- | --- | --- | --- |
| Курс навчання | 2 | 3 | Всього |
| Семестр | 3 | 4 | 5 |
| Основний обсяг часу на предмет | 44 | 56 | 62 | 162 |
| У тому числі аудиторних занять | 32 | 44 | 48 | 124 |
| З них: лекції | 20 | 22 | 28 | 70 |
| Лабораторно-практичні | 12 | 22 | 20 | 54 |
| Консультації | 2 | 2 | 2 | 6 |
| Види індивідуальних завдань |  |  | 5 | 5 |
| Самостійна робота студента | 12 | 12 | 14 | 38 |

**Тематичний план**

з дисципліни

**„Основи програмування та алгоритмічні мови”**

|  |  |  |
| --- | --- | --- |
| №п/п | Назва теми | Обсяг годин за навчальною програмою |
| всього | аудиторні заняття | лекції | лабораторно-практичні заняття | самостійні заняття | контр. роботи |
| 1 | Інтегровані середовища розробки. Microsoft Visual Studio. Microsoft Visual C++. | 16 | 14 | 8 | 4 | 2 |  |
| 2 | Прості типи даних. | 14 | 12 | 8 | 0 | 2 |  |
| 3 | Системи числення. | 14 | 10 | 6 | 2 | 4 |  |
| 4 | Керуючі структури. | 18 | 14 | 8 | 6 | 4 | 2 |
| 5 | Функція як фундаментальне поняття мови С++. | 14 | 10 | 6 | 2 | 4 |  |
| 6 | Генерація випадкових чисел. | 14 | 10 | 6 | 4 | 4 |  |
| 7 | Складені типи даних. Масиви. | 22 | 16 | 8 | 18 | 6 |  |
| 8 | Складені типи даних. Рядки. | 16 | 12 | 6 | 6 | 4 |  |
| 9 | Складені типи даних.Структура. Клас. | 18 | 14 | 8 | 8 | 4 |  |
| 10 | Складені типи даних.Покажчики. | 16 | 12 | 6 | 4 | 4 |  |
| ***Всього*** | ***162*** | ***124*** | ***70*** | ***54*** | ***38*** | *2* |

**Програма дисципліни**

**„Основи програмування та алгоритмічні мови”**

Тема 1. *Інтегровані середовища розробки. Microsoft Visual Studio.
Microsoft Visual C++.*

Історія розвитку мов програмування. Машинні мови. Асемблер. Мови програмування високого рівня. Транслятор. Компілятор. Інтерпретатор.

Опис типового середовища програмування на мові С++. Інтегровані середовища розробки (Integrated Development Environment). Microsoft Visual Studio. Microsoft Visual C++.

Перша програма на мові Visual С++. Початкові відомості про мову С++. Особливості роботи в середовищі С++. Створення найпростіших програм лінійної структури.

Практична робота №1 (2 год.) Перша програма на мові С++.

Практична робота №2. (2 год.) Найпростіші програми лінійної структури.

*Тема 2. Прості типи даних.*

Типи даних. Вивчення цілочислових типів даних. Дійсний тип даних. Числа з плаваючою точкою. Арифметичні операції. Пріоритет операцій та асоціативність. Операція ділення. Операція ділення по модулю. Перетворення типів даних. Перетворення даних у виразах.

*Тема 3. Системи числення.*

Системи числення. Стандартний метод запису чисел. Двійкові числа. Вісімкові числа. Шістнадцяткові числа. Двійкові числа. Метод швидкого переведення чисел з двійкової системи числення у шістнадцяткову і навпаки. Навчальна програма по темі „ Системи числення”. Самостійна робота.

Практична робота №3. (2 год.) Системи числення.

*Тема 4. Керуючі структури.*

Оператори розгалуження. Оператор if, if/else. Оператор switch. Логічні операції. Розв’язування задач.

Реалізація циклічних конструкцій у мові С++. Цикл for. Цикл while. Цикл do while. Порівняння циклів for та while. Оператори break та continue.

Оператори інкремента та декремента. Комбіновані оператори присвоєння.

Практична робота №4. (2 год.) Оператори розгалуження. Оператор if, if/else. Оператор switch. Логічні операції.

Практична робота №5. (2 год.) Реалізація циклічних конструкцій у мові С++. Цикл for. Цикл while. Цикл do while. Порівняння циклів for та while. Оператори break та continue. Оператори інкремента та декремента. Комбіновані оператори присвоєння.

Практична робота №6. (2 год.) Реалізація циклічних конструкцій у мові С++. Розв’язування задач.

*Тема 5. Функція як фундаментальне поняття мови С++.*

Функція як фундаментальне поняття мови С++. Прототип функції. Визначення функції. Виклик функції. Практикум по програмуванню.

Передача функціям аргументів за значенням.

Практична робота №7. (2 год.) Функція як фундаментальне поняття мови програмування С++.

*Тема 6. Генерація випадкових чисел.*

Генерація випадкових чисел. Функція rand() із стандартної бібліотеки cstdlib. Створення ігрового додатку „Гра Крепс”.

Практична робота №8. (2 год.) Генерація випадкових чисел.

Практична робота №9. (2 год.) Генерація випадкових чисел. Практикум по програмуванню.

*Тема 7. Складені типи даних. Масиви.*

Складені типи даних. Масиви. Ініціалізація масивів. Практикум по програмуванню. Сортування масивів. Пошук у масивах. Послідовний пошук. Бінарний пошук. Обчислення середнього значення, медіани, моди з використанням масивів.

Практична робота №10. (2 год.) Масиви. Практикум по програмуванню. Середнє значення всіх елементів цілочислового масиву.

Практична робота №11. (2 год.) Масиви. Практикум по програмуванню. Визначення найбільшого та найменшого елементу в заданому масиві.

Практична робота №12. (2 год.) Масиви. Практикум по програмуванню. Добуток всіх парних елементів масиву.

Практична робота №13. (2 год.) Масиви. Практикум по програмуванню. Двовимірні масиви. Знайти суму всіх елементів масиву, які розміщені по діагоналі.

Практична робота №14. (2 год.) Масиви. Практикум по програмуванню. Кількість елементів між найменшим та найбільшим елементами масиву.

Практична робота №15. (2 год.) Масиви. Практикум по програмуванню. Визначити індекс найбільшого елементу масиву.

Практична робота №16. (2 год.) Масиви. Сортування масивів методом обміну. Сортування масивів методом вибору. Сортування масивів методом вставки.

Практична робота №17. (2 год.) Пошук у масивах. Послідовний пошук.

Практична робота №18. (2 год.) Пошук у масивах. Бінарний пошук.

*Тема 8. Складені типи даних. Рядки.*

Рядки. Конкатенація рядків. Використання рядків у масивах. Цикли та введення текстів. Функція-елемент cin.get(). Функція-елемент cin.getline(). Вправи по програмуванню.

Клас cstring. Робота з рядками на основі бібліотеки класу cstring. Функції для роботи з рядками: append(), find(), length(), replace(), assign(), empty(), erase(),insert(), strcpy().

Практична робота №19. (2 год.) Функція-елемент cin.get(). Функція-елемент cin.getline(). Вправи по програмуванню.

Практична робота №20. (2 год.) Цикли та введення текстів.

Практична робота №21. (2 год.) Функції для роботи з рядками: append(), find(), length(), replace(), assign(), empty(), erase(),insert(), strcpy().

*Тема 9. Складені типи даних. Структура. Клас.*

Структура. Об’єднання. Перерахування. Масиви структур.

Об’єкти і класи. Абстрактні типи даних. Інкапсуляція. Наслідування. Поліморфізм. Абстрагування. Конструктор. Деструктор. Оператор визначення діапазону доступу. Оператор належності.

Практична робота №22. (4 год.) Структура. Об’єднання. Перерахування.

Практична робота №23. (4 год.) Об’єкти і класи. Абстрактні типи даних.

*Тема 10. Складені типи даних. Покажчики.*

Покажчики. Оголошення та ініціалізація покажчиків. Покажчики та числа. Використання покажчиків як засіб збереження адреси. Звернення до даних через покажчики. Пам’ять стекова і динамічно розподільна. Використання оператора new для створення динамічних масивів. Використання оператора new для створення динамічних структур. Покажчики, масиви та арифметика покажчиків.

Практична робота №24. (4 год.) Покажчики. Оголошення та ініціалізація покажчиків.

**Література**

Базова:

1. Стивен Прата „Язык программирования С++. Лекции и упражнения“ Учебник: Пер. с англ. / Стивен Прата – СПб: ООО „ДиаСофтЮП“, 2013 – 1104 с.
2. Вирт Н. Алгоритмы и структуры данных / Пер. с англ. – М.: Мир, 1989. – 360 с., ил.
3. Гринзоу Лу. Философия программирования для Windows 95/NT / Пер. с англ.- СПб.: Символ-Плюс, 1997. – 640 с., ил.
4. Джейс Либерти Освой самостоятельно С++ за 21 день: 3-е изд. пер. с англ.: Уч. пос. – М.: Издательский дом „Вильямс”, 2011. – 816 с.: ил.
5. Зелковиц М., Шоу А., Геннон Дж. Принципы разработки программного обеспечения / Пер. с англ. –М.: Мир, 1982. – 386 с., ил.
6. Ковалюк Т. В. Основи програмування. – К.: Видавнича група BHV, 2013/ - 384 c.: іл.

Допоміжна:

1. Культин Н. Б. С / С++ в примерах и задачах СПб. БХВ – Петербург, 2013. – 228 с., ил.
2. Культин Н. Б. С++ Builder в задачах и примерах. – СПб.: БХВ – Петербург, 2005. – 336 с.: ил.
3. Культин Н. Б. Самоучитель С++ Builder. – СПб.: БХВ – Петербургб 2005. – 320 с.: ил.
4. 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.
5. Прата Стивен Язык программирования С++. Лекции и упражнения. Учебник: Пер. с англ./ Стивен Прата – СПб.:ООО «ДиаСофтЮП», 2003.-1104 с.
6. Страуструп Б. Язык программирования С++ М.: Радио и связь, 1991.
7. Фокс Дж. Программное обеспечение и его разработка / Пер. с англ. – М.: Мир, 1985. – 386 с., ил.
8. Харви Дейтел, Пол Дейтел Как программировать на С++: Третьеиздание. Пер. с англ. – М.: ЗАО «Издательство БИНОМ», 2001 г. – 1152 с.: ил.