




УТВЕРЖДАЮ  
Председатель приемной комиссии

 И.В. Богданов  
«22» октября 2021 г.

## ПРОГРАММА ВСТУПИТЕЛЬНОГО ИСПЫТАНИЯ НА БАЗЕ СПО

### ОСНОВЫ ПРОГРАММИРОВАНИЯ

Программа разработана на основе федерального государственного образовательного стандарта среднего профессионального образования по специальности 09.02.03 ПРОГРАММИРОВАНИЕ В КОМПЬЮТЕРНЫХ СИСТЕМАХ, входящей в состав укрупненной группы направлений подготовки 09.00.00 ИНФОРМАТИКА И ВЫЧИСЛИТЕЛЬНАЯ ТЕХНИКА.

Программа конкретизирует содержание предметных тем образовательного стандарта по разделам курса и рекомендуемую последовательность изучения тем и разделов учебной дисциплины.

Программа составлена на основе требований к уровню подготовки абитуриентов, имеющих среднее профессиональное образование.

Вступительное испытание по *основам программирования* проводится в форме письменного теста. На выполнение работы отводится 60 минут.

Результаты вступительного испытания оцениваются по стобалльной шкале (Приложение).

Абитуриент должен

**уметь:**

- работать в среде программирования;
- реализовывать построенные алгоритмы в виде программ на конкретном языке программирования;

**знать:**

- этапы решения задачи на компьютере;
- типы данных;
- базовые конструкции изучаемых языков программирования;
- принципы структурного и модульного программирования;
- принципы объектно-ориентированного программирования.

#### Раздел 1. Основные принципы алгоритмизации и программирования

1. Основные понятия алгоритмизации: Понятие алгоритма, его свойства и виды. Базовые алгоритмические конструкции: линейная, разветвляющаяся, циклическая. Блок-схема. Логические основы алгоритмизации.

2. Основные принципы программирования: Основы структурного программирования. Развитие языков и технологий программирования. Структура и способы описания языков программирования высокого уровня.

## **Раздел 2. Основы программирования на языке C#**

1. Основные элементы языка: Базовый синтаксис C#. Переменные. Константы. Типы данных. Понятие выражения. Математические операторы. Старшинство операторов.
2. Управляющие структуры: Операторы отношения. Условный оператор. Вложенные условные операторы. Тернарный оператор. Оператор выбора. Операторы перехода. Создание проекта разветвляющейся структуры. Создание проекта с использованием оператора выбора.
3. Циклические структуры: Циклы с параметром, предусловием и постусловием. Вложенные циклы. Операторы выхода из цикла: break, continue, return. Бесконечные циклы. Создание проекта с использованием операторов цикла while, do, for, foreach. Создание проекта с использованием вложенных операторов цикла.
4. Структуры данных: Массивы. Объявление одномерного массива. Обработка одномерный массивов: сортировка и поиск. Двумерные массивы. Ступенчатые массивы. Создание проекта с использованием одномерного массива. Создание проекта с использованием двумерного массива. Создание проекта с использованием методов сортировки. Создание проекта с использованием методов поиска. Создание проекта с использованием массива строк и массива символов.
5. Потоки. Файлы: Понятие потока. Механизм буферизации. Классы библиотеки для работы с потоками. Файлы. Создание проекта с использованием обработки исключительных ситуаций. Создание проекта с использованием текстовых файлов. Создание проекта с двоичных файлов. Создание проекта с использованием копирования файлов. Создание проекта с использованием многопоточных приложений.

## **Раздел 3. Объектно-ориентированное программирование**

1. Основные принципы ООП: Класс и объект: характеристика, определение. Принципы ООП. Модификаторы. Методы: создание, размещение методов. Конструкторы. Инкапсуляция, наследование и полиморфизм. Иерархия классов. Скрытие и перекрытие методов. Виртуальные методы. Создание проекта с использованием текстовых элементов управления и окон сообщений. Создание проекта с использованием диалоговых окон. Создание проекта с использованием компонентов для создания меню. Создание проекта с использованием графических объектов. Создание проекта с использованием многооконного интерфейса. Создание проекта с использованием списков. Создание проекта с использованием переключателей. Создание проекта с использованием текстовых редакторов. Создание проекта с использованием перегрузки методов. Создание проекта с использованием рекурсивных методов. Создание проекта с использованием методов с переменным

количеством аргументов. Создание классов, иерархически связанных между собой. Создание проекта с использованием абстрактных классов. Создание проекта с использованием виртуальных методов. Создание проекта с использованием наследования.

- Интерфейсы: Интерфейсы: назначение, реализация. Работа с объектами через интерфейсы. Работа с объектами через интерфейсы. Стандартные интерфейсы: NET: IComparable. Стандартные интерфейсы: NET: ICloneable. Стандартные интерфейсы: NET: IEnumerable. Создание проекта с использованием интерфейса IComparable. Создание проекта с использованием интерфейса ICloneable. Создание проекта с использованием интерфейса IEnumerable. Создание проекта с использованием операторов is и as. Создание проекта с использованием наследования в интерфейсах.
- Структуры и перечисления: Структуры. Работа со структурами. Enum. Операции с перечислениями. Создание проекта с использованием перечислений. Создание проекта с использованием структур.
- Коллекции. Делегаты. События и потоки выполнения: Коллекции. Класс List. Делегаты. Обработка события: автоматическое создание обработчиков. Создание библиотеки. Решение задач на создание событий. Работа с ArrayList. Создание проекта с использованием делегатов. Создание проекта с использованием групповых делегатов. Создание проекта с использованием обработки события.

## **Литература для подготовки**

### ***Учебные пособия:***

- Биллиг, В.А. Основы объектного программирования на C# (C# 3.0, Visual Studio 2008): учебное пособие / В.А. Биллиг. - Москва, Саратов: Интернет-Университет Информационных Технологий (ИНТУИТ), Вузовское образование, 2017. - 583 с.
- Лубашева, Т.В. Основы алгоритмизации и программирования: учебное пособие / Т.В. Лубашева, Б.А. Железко. - Минск: Республиканский институт профессионального образования (РИПО), 2016. - 379 с.
- Медведев, М.А. Программирование на СИ# : учебное пособие для СПО / М.А. Медведев, А.Н. Медведев; под редакцией А.В. Присяжного. - 2-е изд. - Саратов, Екатеринбург: Профобразование, Уральский федеральный университет, 2019. - 62 с.

### ***Нормативные источники:***

- ГОСТ 19.504-79 ЕСПД. Руководство программиста. Требования к содержанию и оформлению (с Изменением № 1)
- ГОСТ 19.701-90. ЕСПД. Схемы алгоритмов, программ, данных и систем. Обозначения условные и правила выполнения.

***Интернет-ресурсы:***

6. Основы программирования на С# [Электронный ресурс]. - URL: [https://www.intuit.ru/studies/professional\\_skill\\_improvements/1739/info](https://www.intuit.ru/studies/professional_skill_improvements/1739/info). - Режим доступа: свободный
7. Программирование на языке высокого уровня С# [Электронный ресурс] - URL: <https://www.intuit.ru/studies/courses/629/485/info>. - Режим доступа: свободный

### Шкала оценивания

При приеме на обучение по программам бакалавриата результаты каждого вступительного испытания, проводимого Академией самостоятельно, оцениваются по стобалльной шкале.

$$\text{Результат в баллах} = \frac{\text{Количество правильных ответов}}{\text{Количество заданий теста}} * 100$$

где

Результат в баллах – результат вступительного испытания поступающего (по стобалльной шкале).

Количество правильных ответов – количество правильных ответов, данных поступающим при выполнении заданий теста.

Количество заданий теста – количество заданий, которое необходимо выполнить поступающему во время вступительного испытания в соответствии с программой вступительного испытания.

Минимальное количество баллов, подтверждающее успешное прохождение вступительного испытания, определяется локальным актом Академии (Приказ о утверждении перечня вступительных испытаний с указанием приоритетности вступительных испытаний при ранжировании списков поступающих; минимального и максимального количества баллов; информации о формах проведения вступительных испытаний, проводимых Академией самостоятельно)