

Министерство науки и высшего образования Российской Федерации
Муромский институт (филиал)
федерального государственного бюджетного образовательного учреждения высшего образования
**«Владимирский государственный университет
имени Александра Григорьевича и Николая Григорьевича Столетовых»
(МИ ВлГУ)**

Отделение среднего профессионального образования

«УТВЕРЖДАЮ»
Заместитель директора по УР
_____ Д.Е. Андрианов
« 17 » 05 2022 г.

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Основы алгоритмизации и программирования

для специальности 09.02.07 Информационные системы и программирование

Муром, 2022 г.

Рабочая программа учебной дисциплины разработана на основе Федерального государственного образовательного стандарта (далее - ФГОС) по специальности среднего профессионального образования (далее - СПО) 09.02.07 Информационные системы и программирование №1547 от 09 декабря 2016 года.

Кафедра-разработчик: информационных систем.

Рабочую программу составил: к.ф. н., доцент Кулигин Михаил Николаевич

от «26» апреля 2022 г.

(подпись)

Рабочая программа рассмотрена и одобрена на заседании кафедры ИС.

Протокол № 18

от «26» апреля 2022 г.

Заведующий кафедрой ИС *Андреанов Д.Е.*

(подпись)

СОДЕРЖАНИЕ

1. ПАСПОРТ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ	4
2. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ	6
3. УСЛОВИЯ РЕАЛИЗАЦИИ УЧЕБНОЙ ДИСЦИПЛИНЫ	8
4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ	9

1. ПАСПОРТ РАБОЧЕЙ ПРОГРАММЫ УЧЕБНОЙ ДИСЦИПЛИНЫ

Основы алгоритмизации и программирования

1.1. Область применения рабочей программы

Рабочая программа учебной дисциплины является частью программы подготовки специалистов среднего звена в соответствии с ФГОС по специальности 09.02.07 Информационные системы и программирование.

Рабочая программа учебной дисциплины может быть использована в дополнительном профессиональном образовании, для получения дополнительных компетенций, умений и знаний, необходимых для обеспечения конкурентоспособности выпускника на рынке труда и продолжения образования по специальности.

1.2. Место учебной дисциплины в структуре программы подготовки специалистов среднего звена

Дисциплина ОП.04 Основы алгоритмизации и программирования является общепрофессиональной дисциплиной

Дисциплина базируется на школьном курсе информатики и математики, и является базовой для всех специальных дисциплин направления подготовки.

1.3. Цели и задачи учебной дисциплины - требования к результатам освоения учебной дисциплины:

Целью дисциплины является изучение и освоение базовых понятий и приемов программирования, применяемых на всех основных этапах разработки программ; изучение методов программирования для овладения знаниями в области технологии программирования; подготовка к осознанному использованию как языков программирования, так и методов программирования.

Задачами дисциплины является формирование целостного представления о предмете, основание теоретических знаний и практических навыков, позволяющих ориентироваться в области разработки алгоритмов решения задач и написания программных кодов на языке высокого уровня; освоение теоретических основ структурного программирования; изучение конкретных языков программирования; использования языков программирования для решения практических задач

В результате освоения дисциплины обучающийся должен знать:

- этапы решения задачи на компьютере (ПК 1.1.);
- типы данных (ПК 1.1.);
- базовые конструкции изучаемых языков программирования (ПК 1.1.);
- принципы структурного и модульного программирования (ПК 1.1.).

В результате освоения дисциплины обучающийся должен уметь:

- работать в среде программирования (ПК 1.2.);
- реализовывать построенные алгоритмы в виде программы на конкретном языке программирования (ПК 1.2.);
- оформлять код программы в соответствии со стандартом кодирования (ПК 1.1.).

В результате освоения дисциплины обучающийся должен владеть следующими общими (ОК) и профессиональными (ПК) компетенциями:

- ПК 1.1. Формировать алгоритмы разработки программных модулей в соответствии с техническим заданием.;
- ПК 1.2. Разрабатывать программные модули в соответствии с техническим заданием.;
- ПК 1.3. Выполнять отладку программных модулей с использованием специализированных программных средств.;
- ПК 1.4. Выполнять тестирование программных модулей.;

- ПК 2.3. Выполнять отладку программного модуля с использованием специализированных программных средств.;

1.4. Количество часов на освоение программы учебной дисциплины:

Максимальной учебной нагрузки обучающегося 59 часов, в том числе:

обязательной аудиторной нагрузки обучающегося 50 часов;

самостоятельной нагрузки обучающегося 9 часов.

2. СТРУКТУРА И СОДЕРЖАНИЕ УЧЕБНОЙ ДИСЦИПЛИНЫ

2.1. Объем учебной дисциплины и виды учебной работы

Вид учебной работы	Объем часов
	3 семестр
Максимальная учебная нагрузка (всего)	59
Обязательная аудиторная учебная нагрузка (всего)	50
В том числе:	
лекционные занятия	18
практические занятия	16
лабораторные работы	16
контрольные работы	
курсовая работа / индивидуальный проект	0
Самостоятельная работа обучающегося (всего)	9
Итоговая аттестация в форме	Экзамен

2.2. Тематический план и содержание учебной дисциплины

Наименование разделов и тем	Содержание учебного материала, лабораторные работы и практические занятия, самостоятельная работа обучающихся	Объем часов	Уровень освоения
1	2	3	4
	3 семестр		
Раздел 1	Основы алгоритмизации		
Тема 1.1 Основы алгоритмизации	<i>Содержание учебного материала</i>		
	<i>Лекционные занятия.</i> Стадии разработки программного обеспечения. Понятие алгоритма.	1	1
	<i>Самостоятельная работа обучающихся.</i> Нотации блок-схем.	2	3
Раздел 2	Основы программирования на С#		
Тема 2.1 Основы языка С#	<i>Содержание учебного материала</i>		
	<i>Лекционные занятия.</i> Обзор .NET. Основные понятия. Принципы программирования.	1	1
	<i>Практические занятия.</i> Приложения С для расчетов по формулам, консольный ввод-вывод.	2	2
	<i>Самостоятельная работа обучающихся.</i> История развития .NET Framework.	2	3
Тема 2.2 Управляющие операторы	<i>Содержание учебного материала</i>		
	<i>Лекционные занятия.</i> Система типов. Понятие переменной. Константы. Выражения и операторы. Условный оператор if, оператор switch. Операторы цикла.	6	1
	<i>Практические занятия.</i> Использование условного оператора для решения задач.	2	2
	<i>Лабораторные работы.</i> Условный оператор в С#. Разработка алгоритмов с циклическими конструкциями.	8	3
	<i>Самостоятельная работа обучающихся.</i> Расширенные типы данных. Особенности	5	3

	арифметическиз операций.		
Тема 2.3 Массивы в С#	<i>Содержание учебного материала</i>		
	<i>Лекционные занятия.</i> Одномерные массивы. Базовые алгортмы обработки одномерных массивов. Многомерные массивы. Базовые алгортмы обработки многомерных массивов.	4	1
	<i>Практические занятия.</i> Обработка элементов массивов. Поиск элементов в многомерном массиве. Генерация многомерных масивовпо заданному закону.	4	2
	<i>Лабораторные работы.</i> Одномерные массивы в С .	4	3
Тема 2.4 Структурное программирование в С#	<i>Содержание учебного материала</i>		
	<i>Лекционные занятия.</i> Понятие функции, метода. Использование методов. Передача параметров в методы.	4	1
	<i>Практические занятия.</i> Написание простейших функций. Массив в качестве параметра функций. Написание функций с паармерами.	6	2
	<i>Лабораторные работы.</i> Написание простейших функций.	4	3
Тема 2.5 Строки в С#	<i>Содержание учебного материала</i>		
	<i>Лекционные занятия.</i> Понятие строки в С . Обработка строк как масивов.	2	1
	<i>Практические занятия.</i> Решение задач по обработке строк.	2	2
Всего:		59	

3. УСЛОВИЯ РЕАЛИЗАЦИИ УЧЕБНОЙ ДИСЦИПЛИНЫ

3.1. Требования к минимальному материально – техническому обеспечению

Лаборатория системного и прикладного программирования

6 шт. компьютеров Intel Core i5, 3500 MHz/ ОЗУ 6Gb/ SSD-512Gb/ LG 22'; 6 шт. персональных компьютеров Digitech (комплект2) Intel Core i5 3000 MHz/ DDR-4 12Gb/ SSD-512Gb/ Philips 21eb; проектор NEC V300X 3D; экран проекционный настенный Lumien Master Picture; маршрутизатор Gigabit Switch TEG-S16S; макет системы мобильного мониторинга; лабораторный стенд для изучения микроконтроллера; роботизированная платформа IE-POP-BOT; аппаратно-программный комплекс «Изучение принципов построения и исследования инфокоммуникационных локальных сетей». Маркерная доска. Доступ к сети Интернет.

Кабинет стандартизации и сертификации

Ноутбук ASUS A75n 17,1; доступ к сети Интернет; Проектор Acer X127H DLP; экран проекционный настенный DA-LITE 170*210мм. Комплект учебно-наглядных пособий.

Программное обеспечение:

Microsoft Visual Studio (Программа Microsoft Azure Dev Tools for Teaching (Order Number: IM126433))

Adobe Acrobat Reader DC (Общие условия использования продуктов Adobe)

3.2. Информационное обеспечение обучения

Перечень рекомендуемых учебных изданий, интернет – ресурсов, дополнительной литературы.

Основные источники:

1. Журавлева, М. Г. Основы программирования. Введение в язык Си. Ч.1 : учебное пособие по курсам «Программирование», «Основы алгоритмизации и программирования» / М. Г. Журавлева, В. А. Алексеев, П. А. Домашнев. — Липецк : Липецкий государственный технический университет, ЭБС АСВ, 2019. — 99 с. — ISBN 978-5-00175-001-7. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/101463.html> (дата обращения: 21.02.2023). — Режим доступа: для авторизир. пользователей. <https://www.iprbookshop.ru/101463.html>
2. Марченко, А. Л. Основы программирования на C# 2.0 : учебное пособие / А. Л. Марченко. — 3-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2020. — 551 с. — ISBN 978-5-4497-0680-5. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: <http://www.iprbookshop.ru/97566.html> (дата обращения: 20.12.2020). — Режим доступа: для авторизир. пользователей. <http://www.iprbookshop.ru/97566.html>

Дополнительные источники:

1. Забержинский, Б. Э. Программирование. Введение в разработку на C# : учебное пособие / Б. Э. Забержинский, А. Г. Золин. — Самара : Самарский государственный технический университет, ЭБС АСВ, 2017. — 120 с. — Текст : электронный // Цифровой образовательный ресурс IPR SMART : [сайт]. — URL: <https://www.iprbookshop.ru/90876.html> (дата обращения: 21.02.2023). — Режим доступа: для авторизир. пользователей. <http://www.iprbookshop.ru/90876.html>

Интернет-ресурсы:

1. Электронная библиотека ВлГУ - <http://e.lib.vlsu.ru/>
2. электронная библиотечная системы "IPRBooks" (<http://www.iprbookshop.ru/>)

4. КОНТРОЛЬ И ОЦЕНКА РЕЗУЛЬТАТОВ ОСВОЕНИЯ УЧЕБНОЙ ДИСЦИПЛИНЫ

Результаты обучения (освоенные умения, усвоенные знания)	Формы и методы контроля и оценки результатов обучения
работать в среде программирования	Индивидуальный контроль при выполнении и защите лабораторных работ и заданий по практике.
реализовывать построенные алгоритмы в виде программы на конкретном языке программирования	Индивидуальный контроль при выполнении и защите лабораторных работ и заданий по практике.
оформлять код программы в соответствии со стандартом кодирования	Индивидуальный контроль при выполнении и защите лабораторных работ и заданий по практике.
этапы решения задачи на компьютере	Индивидуальный контроль при выполнении и защите лабораторных работ и заданий по практике.
типы данных	Индивидуальный контроль при выполнении и защите лабораторных работ и заданий по практике.
базовые конструкции изучаемых языков программирования	Индивидуальный контроль при выполнении и защите лабораторных работ и заданий по практике.
принципы структурного и модульного программирования	Индивидуальный контроль при выполнении и защите лабораторных работ и заданий по практике.

Фонд оценочных материалов (средств) по дисциплине
Основы алгоритмизации и программирования

1. Оценочные материалы для проведения текущего контроля успеваемости по дисциплине

Верно ли написание в программе следующей строки для вывода текста на консоль?
`Console.WriteLine("Hello, World!");`

- Верно
- Не верно
- Нужно всегда писать `System.Console.WriteLine("Hello, World!");`
- Верно при условии, если выше написано `using System;`

Дан массив `myArr`. Требуется в переменную `A` записать его длину. Как это сделать?

- `A = myArr.Length`
- `myArr.Length = A`
- `A = myArr.Length`
- `A = max(myArr)`
- `A = myArr(i)`

Как найти квадратный корень из числа `x`

- `Sqrt(x)`
- `Summ.Koren(x);`
- `Arifmetic.sqrt(x);`
- `Math.Sqrt(x);`

Какой класс в `C#` предоставляет возможности по считыванию из файла и записи в файл

- `FileStream`
- `FileReadWrite`
- `FileOutputInput`
- `FileDirect`

Какой язык программирования Вы использовали на лабораторных работах?

- Pascal
- C#
- C++
- Visual Studio
- Английский
- Процедурный

Категория объектов, обладающих одинаковыми свойствами и поведением.

- Класс
- Объект
- Модуль
- Подпрограмма

Компилятор выполняет следующие основные задачи:

- Выполняет анализ и проверяет синтаксис транслируемой программы, и, в случае обнаружения синтаксической ошибки, прерывает свою работу и выдает соответствующее сообщение.
- Генерирует машинный код программы
- Проверяет правильность алгоритма транслируемой программы, и, в случае обнаружения синтаксической ошибки, прерывает свою работу и выдает соответствующее сообщение.

- Генерирует код программы на выбранном языке программирования

Объект это:

- Это экземпляр какого-либо класса
- Часть подпрограммы
- Синоним слова "переменная"
- Основная программа

Процедура, которая определена как часть класса и инкапсулирована (содержится) в нем
это

- Метод
- Объект
- Функция
- Модуль

Укажите НЕверный способ объявления массивов

- `int[] array = new int[5];`
- `string[] stringArray = new string[6];`
- `int[] MASSIV = new int[] { 1, 3, 5, 7, 9 };`
- Все варианты верные
- `string[] array = string[0..6,0..6];`

Чему будет равен c, если `int a = 10; int b = 4; int c = a % b;`

- 11
- 2
- 3
- 1

Что делает оператор `"%"`

- Возвращает процент от суммы
- Возвращает остаток от деления
- Возвращает тригонометрическую функцию
- Ни чего из выше перечисленного.

Это средство записи инструкций программы простыми командами на аппаратном уровне. Такие языки зависят от набора команд конкретной ЭВМ.

- Языки низкого уровня
- Языки высокого уровня
- Языки среднего уровня
- Командные языки

Это средство записи программы в наглядном, легко воспринимаемом виде. Каждый такой язык ориентирован не на систему команд конкретной ЭВМ, а на систему инструкций, характерных для записи алгоритмов определенного класса.

- Языки высокого уровня
- Языки низкого уровня
- Языки среднего уровня
- Алгоритмические языки

Общее распределение баллов текущего контроля по видам учебных работ для студентов

Рейтинг-контроль 1	Защищённые отчеты по лабораторным работам, выполненные практические задания	18 баллов
Рейтинг-контроль 2	Защищённые отчеты по лабораторным работам, выполненные практические задания	18 баллов
Рейтинг-контроль 3	Защищённые отчеты по лабораторным работам, выполненные практические задания	18 баллов
Посещение занятий студентом		6 баллов
Дополнительные баллы (бонусы)		
Выполнение семестрового плана самостоятельной работы		

2. Промежуточная аттестация по дисциплине

Перечень вопросов к экзамену / зачету / зачету с оценкой.

Перечень практических задач / заданий к экзамену / зачету / зачету с оценкой (при наличии)

В C# различие в регистре букв:

- Имеет значение
- Не имеет значения
- Имеет значение только при написании переменных
- Имеет значение только при написании функций
- Имеет значение только в старых версиях VisualStudio

Вещественный тип данных объявляется служебным словом (C#):

- int
- integer
- char
- string
- str
- symbol
- double

В каком случае переменная A будет содержать квадратный корень числа B? (C#)

- A=Math.Sqrt(B);
- A:=Sqrt(B);
- A=Sqr(B);
- A==Sqrt(B).Math;
- Char A=Sqrt(B);

В каком случае переменная num будет равна 1? (C#)

- num = 13 / 3;
- num = 13 % 3;

- num = 13 div 3;
- num = 13 sqrt(3);
- num = length(3);

В программе на языке C# что означает строка "using System;"?

- в программе используется пространство имен System
- Программа называется System
- Программа предназначена для системных операций
- В программе обязательно будут использоваться файловые операции

Глобальные переменные в программе являются

- Статическими переменными
- Динамическими переменными
- Полустатическими переменными
- Независимыми переменными

Данные относят к этому типу, если в любой момент времени они определяют одно и только одно значение

- Простой тип
- Целый тип
- Числовой тип
- Однородный тип

До каких пор будут выполняться операторы в теле цикла while (x == 100)?

- Пока x равен ста
- Пока x строго меньше ста
- Пока x больше ста
- Пока x меньше или равен ста

Какая из операций определяет остаток целочисленного деления? (C#)

- div
- mod
- in
- or
- %
- *
- /

Какая из операций является логической? (C#)

- div
- mod
- in
- or
- |
- +

Какая связка обозначает цикл с известным числом повторений?

- For...{...}
- if...else
- Case...of... end
- While...{...}
- For...count...

Какая связка операторов обозначает цикл с выходом по условию?

- For...{...}
- if...else
- Case...of... end
- While...{...}
- For...if...exit

Какие служебные символы используются для обозначения начала и конца блока кода?
(C#)

- ()
- /* */
- begin end
- { }

Как на языке C# будет записан оператор "не равно"?

- ==
- !=
- not =
- !

Какой из следующих операторов - оператор сравнения двух переменных? (C#)

- ==
- equal
- :=
- =
- In
- if

Локальные переменные подпрограмм являются:

- Статическими переменными
- Динамическими переменными
- Полустатическими переменными
- Независимыми переменными

Место в памяти для размещения статических переменных определяется:

- При компиляции программы
- При выполнении программы
- Во время написания кода программы
- В параметрах компилятора

Ограниченная упорядоченная совокупность однотипных величин это:

- Массив
- Множество
- Запись
- Строка

Оператор инкрементации (x=x+1) (C#) эквивалентен...

- X++
- X+
- Inc(x)
- X.inc

Описанная явным образом в программе переменная, обращение к которой осуществляется по имени.

- Статическая переменная

- Динамическая переменная
- Структурированная переменная
- Производная переменная

Перевод программ с языка высокого уровня на язык более низкого уровня обеспечивает программа:

- ассемблер
- паскаль
- компилятор
- c#
- c++
- windows

Последовательность операторов, которые определены и записаны только в одном месте программы, однако их можно вызвать для выполнения из одной или нескольких точек программы

- Подпрограмма
- Цикл
- Модуль
- Объект

По структурному признаку данные делятся на :

- Числовые
- Однородные
- Неоднородные
- Текстовые
- Дробные
- Графические

Программа, переводящая текст программы поэтапно (покомандно) и сразу же выполняющая оттранслированную команду программы.

- Интерпритатор
- Компилятор
- Транслятор
- Ассемблер

Символьный тип данных объявляется служебным словом (с#):

- int
- integer
- char
- string
- str
- symbol

Способы задания алгоритма:

- Графический
- Блок-схема
- Табличный
- Аналитический
- Программный
- Трассировочный

Структуры, для работы с которыми используются указатели в явном и неявном виде

это

- Динамические структуры данных
- Статические структуры данных
- Виртуальные структуры данных
- Файловые структуры данных

Укажите верное утверждение:

- При записи в файл вся информация сначала направляется в буфер и там накапливается до тех пор, пока весь объем буфера не будет заполнен. Только после этого или после специальной команды сброса происходит передача данных на внешнее устройство.
- При записи в файл вся информация передается на внешнее устройство. Если внешнее устройство недоступно, то информация направляется в буфер и там накапливается до тех пор, пока весь объем буфера не будет заполнен.
- При записи в файл вся информация сначала направляется в буфер и там накапливается до тех пор, пока весь объем буфера не будет заполнен. Если внешнее устройство недоступно, то размер буфера динамически увеличивается
- При записи в файл вся информация сначала направляется в буфер и и одновременно направляется на внешнее устройство. Буфе требуется для резервирования информации

Укажите вещественные типы данных

- Float
- Double
- String
- Word
- Byte

Укажите правильное описание переменной строкового типа на языке C#

- Int S="строка"
- String S;
- String S="строка";
- Var S:string;
- Str=S;

Укажите символьные типы данных

- Char
- String
- Word
- Extended

Укажите целочисленные типы данных

- Int
- Sbyte
- Double
- String
- float

У типа byte разрядность - 8 бит. Значения из какого диапазона может принимать переменная этого типа?

- 0..255
- 1..256
- 1..8
- 1..1024
- 1..1000

У типа ushort разрядность - 16 бит. Значения из какого диапазона может принимать переменная этого типа?

- 1..16
- 0..16
- 0..16000
- $0..2^{10}$
- $0..2^{16}$
- $0..(2^{16})-1$

Что такое константа ?

- Переменная типа string
- Переменная которая может быть изменена в любое время.
- Глобальная переменная
- Переменная значение которой нельзя изменить.

Эта программа транслирует весь текст программы с языка программирования в машинный код.

- Интерпритатор
- Компилятор
- Транслятор
- Ассемблер

Этот тип описывается путем задания двух констант, определяющих границы допустимых для данных типов значений

- Интервальный тип
- Перечисляемый тип
- Вещественный тип
- Целый тип

Этот тип представляет собой ограниченную упорядоченную последовательность скалярных констант, составляющих данный тип

- Перечисляемый тип
- Вещественный тип
- Файловый тип
- Тип множество

Методические материалы, характеризующих процедуры оценивания

Выполняется проверка уровня выполнения индивидуальных заданий в рамках практических и лабораторных работ.

Для оценки уровня теоретических и практических знаний используется контрольный устный или письменный опрос студентов по тематике предшествующих лекционных и практических занятий, выполняются и защищаются в форме устного опроса и наглядной демонстрации лабораторные работы. Итоговым средством оценки уровня знаний по курсу является Экзамен, который проводится в устной форме (в форме собеседования) на основании перечня контрольных вопросов и практических заданий по данной дисциплине.

Самостоятельная работа включает подготовку к практическим и лабораторным занятиям, к рейтинг- контрольным работам.

Максимальная сумма баллов, набираемая студентом по дисциплине равна 100.

Оценка в баллах	Оценка по шкале	Обоснование	Уровень сформированности компетенций
Более 80	«Отлично»	Содержание курса освоено полностью, без пробелов, необходимые практические навыки работы с освоенным материалом сформированы, все предусмотренные программой обучения учебные задания выполнены, качество их выполнения оценено числом баллов, близким к максимальному	Высокий уровень
66-80	«Хорошо»	Содержание курса освоено полностью, без пробелов, некоторые практические навыки работы с освоенным материалом сформированы недостаточно, все предусмотренные программой обучения учебные задания выполнены, качество выполнения ни одного из них не оценено минимальным числом баллов, некоторые виды заданий выполнены с ошибками	Продвинутый уровень
50-65	«Удовлетворительно»	Содержание курса освоено частично, но пробелы не носят существенного характера, необходимые практические навыки работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий, возможно, содержат ошибки	Пороговый уровень
Менее 50	«Неудовлетворительно»	Содержание курса не освоено, необходимые практические навыки работы не сформированы, выполненные учебные задания содержат грубые ошибки	Компетенции не сформированы

3. Задания в тестовой форме по дисциплине

Примеры заданий:

В с# различие в регистре букв:

A) Имеет значение

B) Не имеет значения

C) Имеет значение только при написании переменных

D) Имеет значение только при написании функций

E) Имеет значение только в старых версиях VisualStudio

Вещественный тип данных объявляется служебным словом (с#):

- A) int
- B) integer
- C) char
- D) string
- E) str
- F) symbol
- G) double

В каком случае переменная A будет содержать квадратный корень числа B? (с#)

- A) A=Math.Sqrt(B);
- B) A:=Sqrt(B);
- C) A=Sqr(B);
- D) A==Sqrt(B).Math;
- E) Char A=Sqrt(B);

В каком случае переменная num будет равна 1? (C#)

- A) num = 13 / 3;
- B) num = 13 % 3;
- C) num = 13 div 3;
- D) num = 13 sqrt(3);
- E) num = length(3);

В программе используется одномерный целочисленный массив A с индексами от 0 до 10. Ниже представлен фрагмент программы, в котором значения элементов сначала задаются, а затем меняются.

- A) 10 9 8 7 6 5 4 3 2 1 0
- B) 0 1 2 3 4 5 6 7 8 9 10
- C) 10 9 8 7 6 5 6 7 8 9 10
- D) 0 1 2 3 4 5 4 3 2 1 0

Эта программа транслирует весь текст программы с языка программирования в машинный код.

Этот тип описывается путем задания двух констант, определяющих границы допустимых для данных типов значений

Этот тип представляет собой ограниченную упорядоченную последовательность скалярных констант, составляющих данный тип

Этот тип представляет собой ограниченную упорядоченную последовательность скалярных констант, составляющих данный тип

Полный перечень тестовых заданий с указанием правильных ответов, размещен в банке вопросов на информационно-образовательном портале института по ссылке <https://www.mivlgu.ru/iop/question/edit.php?courseid=3248&recurse=0&showhidden=0&qbshowtext=0&cat=42350%2C141108>

Оценка рассчитывается как процент правильно выполненных тестовых заданий из их общего числа.