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)

Кафедра *ЭиВТ*

«УТВЕРЖДАЮ»
Заместитель директора по УР
_____Д.Е. Андрианов
04.06.2019

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

Программирование

Направление подготовки *09.03.01 Информатика и вычислительная техника*

Профиль подготовки *Вычислительные машины, комплексы, системы и сети*

Семестр	Трудоемкость, час./зач. ед.	Лекции, час.	Практические занятия, час.	Лабораторные работы, час.	Консультация, час.	Контроль, час.	Всего (контактная работа), час.	СРС, час.	Форма промежуточного контроля (экз., зач., зач. с оц.)
1	108 / 3	16	16	16	3,6	0,35	51,95	29,4	Экз.(26,65)
2	72 / 2		20	16		2,25	38,25	33,75	Зач.
3	180 / 5	18		32	3,8	0,35	54,15	99,2	Экз.(26,65)
Итого	360 / 10	34	36	64	7,4	2,95	144,35	162,35	53,3

Муром, 2019 г.

1. Цель освоения дисциплины

Целью изучения дисциплины является приобретение обучающимися знаний об основных понятиях теории алгоритмизации и программирования, усвоение, основных принципов и методов программирования.

Задачи дисциплины:

- освоение основ теории алгоритмизации и программирования;
- освоение основных методов и технологий программирования;
- получение практических навыков в разработке программ на языках высокого уровня.

2. Место дисциплины в структуре ОПОП ВО

Курс базируется на знаниях, полученных обучающимися по следующим дисциплинам: «Информатика», «Математическая логика и теория алгоритмов». Базирующимися дисциплинами являются: «Инженерная и компьютерная графика», «Базы данных».

3. Планируемые результаты обучения по дисциплине

Планируемые результаты обучения по дисциплине, соотнесенные с планируемыми результатами освоения ОПОП (компетенциями и индикаторами достижения компетенций)

Формируемые компетенции (код, содержание компетенции)	Планируемые результаты обучения по дисциплине, в соответствии с индикатором достижения компетенции		Наименование оценочного средства
	Индикатор достижения компетенции	Результаты обучения по дисциплине	
ОПК-3 Способен решать стандартные задачи профессиональной деятельности на основе информационной и библиографической культуры с применением информационно-коммуникационных технологий и с учетом основных требований информационной безопасности;	ОПК-3.3 Производит разработку структур данных, алгоритмов и оценку их сложности для решения поставленной задачи.	Знать виды структур данных, алгоритмов и методы оценки их сложности для решения поставленной задачи (ОПК-3.3) Уметь разрабатывать структуры данных, алгоритмы и оценивать их сложность (ОПК-3.3) Владеть наавыком разработки структур данных, алгоритмов и оценки их сложности (ОПК-3.3)	вопросы к устному опросу, задачи
ОПК-8 Способен разрабатывать алгоритмы и программы, пригодные для практического применения;	ОПК-8.1 Определяет суть поставленной задачи и знакомится с особенностями предметной области.	Знать этапы процесса разработки программы для ЭВМ, моделирования и выявление особенностей предметной области для разработки программы решения задачи (ОПК-8.1) Уметь определять суть поставленной задачи и выявлять особенности предметной области для разработки программ (ОПК-8.1) Владеть навыком определения сути поставленной задачи и выявлением особенностей предметной област для разработки программ (ОПК-8.1)	вопросы к устному опросу, задачи

	ОПК-8.2 Выбирает метод решения задачи и разрабатывает алгоритм.	Знать методы разработки программ и алгоритмов решения задач на ЭВМ (ОПК-8.2) Уметь выбирать метод решения задачи и разрабатывать алгоритмы программы (ОПК-8.2) Владеть навыком применения методов решения задачи и разработки алгоритмов программ (ОПК-8.2)	
	ОПК-8.3 Разрабатывает программу в одной из сред программирования.	Знать принципы разработки программ в современных средах программирования (ОПК-8.3) Уметь разрабатывать программу в одной из сред программирования (ОПК-8.3) Владеть языком программирования и навыком разрабатывать программы в средах программирования (ОПК-8.3)	
	ОПК-8.4 Выполняет тестирование программы	Знать методы тестирования программ (ОПК-8.4) Уметь выполнять тестирование программы (ОПК-8.4) Владеть навыками отладки и тестирования программ (ОПК-8.4)	

4. Структура и содержание дисциплины

Общая трудоемкость дисциплины составляет 10 зачетных единиц, 360 часов.

4.1. Форма обучения: очная

Уровень базового образования: среднее общее.

Срок обучения 4г.

4.1.1. Структура дисциплины

№ п\п	Раздел (тема) дисциплины	Семестр	Контактная работа обучающихся с педагогическим работником							Самостоятельная работа	Форма текущего контроля успеваемости (по неделям семестра), форма промежуточной аттестации(по семестрам)
			Лекции	Практические занятия	Лабораторные работы	Контрольные работы	КП / КР	Консультация	Контроль		
1	Введение. Алгоритмизация.	1	2	4							устный опрос, отчет по практическим работам
2	Программирование на ЯВУ.	1	14	12	16					29,4	устный опрос, отчеты по лабораторным и практическим работам
Всего за семестр		108	16	16	16			3,6	0,35	29,4	Экз.(26,65)
3	Программирование на ЯВУ (2 сем.)	2		20	16					33,75	устный опрос, отчет по лабораторным и практическим работам, проверка хода выполнения курсовой работы
Всего за семестр		72		20	16		+	0	2,25	33,75	Зач.
4	Программирование на ЯВУ (3 сем.)	3	18		32					99,2	устный опрос, отчет по лабораторным работам
Всего за семестр		180	18		32			3,8	0,35	99,2	Экз.(26,65)
Итого		360	34	36	64			7,4	2,95	162,35	53,3

4.1.2. Содержание дисциплины

4.1.2.1. Перечень лекций

Семестр 1

Раздел 1. Введение. Алгоритмизация.

Лекция 1.

Введение. Основные этапы решения задач на ЭВМ. Основы алгоритмизации. Способы записи алгоритма (2 часа).

Раздел 2. Программирование на ЯВУ.

Лекция 2.

Язык программирования высокого уровня Object Pascal. Начальные сведения о языке (2 часа).

Лекция 3.

ЯВУ Object Pascal. Структурные типы. Приведение типов. Основные языковые конструкции (2 часа).

Лекция 4.

Событийная модель ОС Windows.Среда программирования Delphi. Создание программы в Delphi. Директивы компилятора. Принципы визуального программирования. Свойства в Delphi. События в Delphi (2 часа).

Лекция 5.

Типы. Константы. Переменные. Множества. Записи. Битовая арифметика (2 часа).

Лекция 6.

Организация подпрограмм. Процедуры и функции (2 часа).

Лекция 7.

Модульное программирование. Модули. Модульные программы (2 часа).

Лекция 8.

Файлы. Файловый тип. Последовательный и прямой доступ к компонентам файла (2 часа).

Семестр 3

Раздел 4. Программирование на ЯВУ (3 сем.)

Лекция 9.

Объектно-ориентированное программирование. Классы. Состав класса (2 часа).

Лекция 10.

Поля класса. Методы класса. Типы методов. Конструкторы и деструкторы. Свойства в классе. Передача сообщений в классах (2 часа).

Лекция 11.

Объектно-ориентированное визуальное программирование. Компоненты. Библиотеки элементов управления в системах программирования. Понятие компонента в RAD Studio. Создание компонентов. Пакеты компонентов (2 часа).

Лекция 12.

Технология COM. Понятие интерфейса. Связь между классами и интерфейсами. COM-сервер. Язык описания интерфейсов IDL. Программирование COM-объектов (2 часа).

Лекция 13.

Технология OLE Automation. Реализация OLE Automation в языках программирования. Тип данных Variant. Программирование OLE-объектов. Элементы управления ActiveX. Программирование ActiveX-объектов (2 часа).

Лекция 14.

Язык программирования C#. Платформа .NET Framework. Среда исполнения CLR. Библиотека классов .NET. Библиотека WindowsForms. Средства разработки на языке C#. Программирование на C# в Microsoft Visual Studio (2 часа).

Лекция 15.

Язык программирования C#. Синтаксис языка C#. Типы данных. Операторы и выражения. Классы в C# (2 часа).

Лекция 16.

Язык программирования Java. Виртуальная машина JavaVM. Среда выполнения программ JRE. JAR-пакеты. Средства разработки на языке Java (2 часа).

Лекция 17.

Язык программирования Java. Синтаксис языка Java. Типы данных. Операторы и выражения. Классы в Java (2 часа).

4.1.2.2. Перечень практических занятий

Семестр 1

Раздел 1. Введение. Алгоритмизация.

Практическое занятие 1

Способы описания алгоритмов. Ветвление (2 часа).

Практическое занятие 2

Циклические алгоритмы (2 часа).

Раздел 2. Программирование на ЯВУ.

Практическое занятие 3

Операции над последовательностями. Вектора (2 часа).

Практическое занятие 4

Массивы (2 часа).

Практическое занятие 5

Символьные массивы (строки) (2 часа).

Практическое занятие 6

Двухмерные массивы (2 часа).

Практическое занятие 7

Алгоритмы сортировки массивов (2 часа).

Практическое занятие 8

Рекурсивные алгоритмы (2 часа).

Семестр 2

Раздел 3. Программирование на ЯВУ (2 сем.)

Практическое занятие 9

Тематика курсовой работы. Графические примитивы. График функции (2 часа).

Практическое занятие 10

Работа с графикой. Игровое поле (2 часа).

Практическое занятие 11

Отрисовка изображения на основе таймера (2 часа).

Практическое занятие 12

Алгоритмы подготовки элементов игрового поля (2 часа).

Практическое занятие 13

Алгоритмы работы с файлами в игровом приложении (2 часа).

Практическое занятие 14

Способы воспроизведения звука (2 часа).

Практическое занятие 15

Алгоритмы обработки действий пользователя в игровом приложении (2 часа).

Практическое занятие 16

Сканирование иерархической структуры директорий по рекурсивному алгоритму (2 часа).

Практическое занятие 17

Сканирование иерархической структуры директорий по рекурсивному алгоритму (2 часа).

Практическое занятие 18

Защита курсовой работы (2 часа).

4.1.2.3. Перечень лабораторных работ

Семестр 1

Раздел 2. Программирование на ЯВУ.

Лабораторная 1.

Изучение процесса создания программы на ЯВУ. Среда программирования Delphi (4 часа).

Лабораторная 2.

Массивы. Работа с числовыми и строковыми массивами. Двухмерные массивы (матрицы). Подпрограммы. Создание подпрограмм обработки массивов (4 часа).

Лабораторная 3.

Процедуры и функции. Вектор. Строка (4 часа).

Лабораторная 4.

Процедуры и функции. Матрица (4 часа).

Семестр 2

Раздел 3. Программирование на ЯВУ (2 сем.)

Лабораторная 5.

Файлы. Запись информации в файлы (4 часа).

Лабораторная 6.

Неоднородные структуры данных. Тип запись в языках программирования (4 часа).

Лабораторная 7.

Динамические структуры данных. Классы (4 часа).

Лабораторная 8.

Наследование элементов класса в Delphi (4 часа).

Семестр 3

Раздел 4. Программирование на ЯВУ (3 сем.)

Лабораторная 9.

Программирование свойств и обработчиков событий класса. Создание графического класса в Delphi (4 часа).

Лабораторная 10.

Создание класса визуального графического элемента управления. Создание компонента в Delphi. Создание пакета компонент (4 часа).

Лабораторная 11.

Создание и программирование ActiveX-компонента. Изучение способов установки и использования объектов ActiveX (4 часа).

Лабораторная 12.

Изучение технологий COM и OLE Automation. Программирование COM-объектов. Создание программы передачи данных и генерации документов для пакета Microsoft Office (4 часа).

Лабораторная 13.

Изучение процесса программирования на языке C#. Программирование на языке C# в среде VisualStudio. Создание программы с графическим интерфейсом на языке C# с использованием объектов WindowsForms (4 часа).

Лабораторная 14.

Разработка классов в языке C#. Создание программы с использованием разработанного класса (4 часа).

Лабораторная 15.

Изучение процесса программирования на языке Java. Среда программирования NetBeans. Создание программы с графическим интерфейсом на языке Java (4 часа).

Лабораторная 16.

Разработка классов в языке Java Создание программы с использованием разработанного класса (4 часа).

4.1.2.4. Перечень тем и учебно-методическое обеспечение самостоятельной работы

Перечень тем, вынесенных на самостоятельное изучение:

1. Критерии качества программы. Жизненный цикл программы.
2. Постановка задачи и спецификация программы.
3. Составные части программного проекта.
4. Языки структурного программирования.
5. Корректность программ. Утверждения о программах.
6. Способы описания подпрограмм. Передача параметров.
7. Варианты представления массивов. Утверждения о массивах.
8. Индуктивные функции на последовательностях (файлах, массивах).
9. Программирование динамических структур в ЯВУ.
10. Способы конструирования и верификации программ.
11. Использование визуальных компонентов при разработке приложений.
12. Использование невизуальных компонентов при разработке приложений.

Для самостоятельной работы используются методические указания по освоению дисциплины и издания из списка приведенной ниже основной и дополнительной литературы.

4.1.2.5. Перечень тем контрольных работ, рефератов, ТР, РГР, РПР

Не планируется.

4.1.2.6. Примерный перечень тем курсовых работ (проектов)

1. Приложения ОС Windows. Технические расчеты.
2. Приложения ОС Windows. Экранные компьютерные игры.
3. Приложения ОС Windows. Логические компьютерные игры.
4. Приложения ОС Windows. Организация документации.
5. Приложения ОС Windows. Текстовые/графические редакторы.

4.2 Форма обучения: заочная

Уровень базового образования: среднее общее.

Срок обучения 5л.

Семестр	Трудоемкость, час./ зач. ед.	Лекции, час.	Практические занятия, час.	Лабораторные работы, час.	Консультация, час.	Контроль,час.	Всего (контактная работа), час.	СРС, час.	Форма промежуточного контроля (экз., зач., зач. с оц.)
1	**90 / 2,5**	**8**		**8**	**4**	**0,6**	**20,6**	**60,75**	**Экз.(8,65)**
2	**90 / 2,5**	**8**	**8**	**8**	**4**	**2,25**	**30,25**	**56**	**Зач.(3,75)**
3	**180 / 5**	**8**		**16**	**4**	**0,6**	**28,6**	**142,75**	**Экз.(8,65)**
Итого	**360 / 10**	**24**	**8**	**32**	**12**	**3,45**	**79,45**	**259,5**	**21,05**

4.2.1. Структура дисциплины

№ п\п	Раздел (тема) дисциплины	Семестр	Контактная работа обучающихся с педагогическим работником							Самостоятельная работа	Форма текущего контроля успеваемости (по неделям семестра), форма промежуточной аттестации(по семестрам)
			Лекции	Практические занятия	Лабораторные работы	Контрольные работы	КП / КР	Консультация	Контроль		
1	Введение. Алгоритмизация.	1	4								устный опрос
2	Программирование на ЯВУ	1	4		8					60,75	устный опрос, отчет по лабораторным работам, проверка хода выполнения контрольной работы
Всего за семестр		90	8		8	+		4	0,6	60,75	Экз.(8,65)
3	Программирование на ЯВУ (2 сем.)	2	8	8	8					56	устный опрос, отчет по лабораторным работам и практическим работам, проверка хода выполнения курсовой работы

Всего за семестр		90	8	8	8		+	4	2,25	56	Зач.(3,75)
4	Программирование на ЯВУ (3 сем.)	3	8		16					142,75	устный опрос, отчет по лабораторным работам, проверка хода выполнения контрольной работы
Всего за семестр		180	8		16	+		4	0,6	142,75	Экз.(8,65)
Итого		360	24	8	32			12	3,45	259,5	21,05

4.2.2. Содержание дисциплины

4.2.2.1. Перечень лекций

Семестр 1

Раздел 1. Введение. Алгоритмизация.

Лекция 1.

Основные этапы решения задач на ЭВМ. Жизненный цикл программы. Алгоритмизация. Способы записи алгоритма (2 часа).

Лекция 2.

Языки высокого уровня Object Pascal и C++. Основные понятия языков программирования. Структура программы (2 часа).

Раздел 2. Программирование на ЯВУ

Лекция 3.

ЯВУ Delphi и C++: Выражения. Типы данных (2 часа).

Лекция 4.

ЯВУ Delphi и C++: Константы. Переменные. Операторы. Подпрограммы. Модули (2 часа).

Семестр 2

Раздел 3. Программирование на ЯВУ (2 сем.)

Лекция 5.

Динамические переменные. Динамические структуры данных. Динамические массивы. Стеки. Очереди. Линейные списки: основные виды и способы реализации; линейный список как абстрактный тип данных (2 часа).

Лекция 6.

Модульные программы. Статические и динамические модули. Создание и использование динамически подключаемых библиотек ОС Windows (2 часа).

Лекция 7.

Объектно-ориентированное программирование. Классы. Состав класса: Поля класса. Методы класса. Типы методов. Конструкторы и деструкторы. Свойства в классе (2 часа).

Лекция 8.

Объектно-ориентированное визуальное программирование. Компоненты. Библиотеки элементов управления в системах программирования (2 часа).

Семестр 3

Раздел 4. Программирование на ЯВУ (3 сем.)

Лекция 9.

Технология COM. Понятие интерфейса. Связь между классами и интерфейсами. COM-сервер. Язык описания интерфейсов IDL. Программирование COM-объектов (2 часа).

Лекция 10.

Технология OLE Automation. Реализация OLE Automation в языках программирования. Тип данных Variant. Программирование OLE-объектов (2 часа).

Лекция 11.

Язык программирования C#. Платформа .NET Framework. Синтаксис языка C#. Типы данных. Операторы и выражения. Классы в C# (2 часа).

Лекция 12.

Язык программирования Java. Виртуальная машина JavaVM. Среда выполнения программ JRE. Синтаксис языка Java. Типы данных. Операторы и выражения. Классы в Java (2 часа).

4.2.2.2. Перечень практических занятий

Семестр 2

Раздел 3. Программирование на ЯВУ (2 сем.)

Практическое занятие 1.

Тематика курсовой работы. Графические примитивы. График функции (2 часа).

Практическое занятие 2.

Работа с графикой. Алгоритмы формирования изображения на основе таймера (2 часа).

Практическое занятие 3.

Алгоритмы обработки действий пользователя в приложении (2 часа).

Практическое занятие 4.

Защита курсовой работы (2 часа).

4.2.2.3. Перечень лабораторных работ

Семестр 1

Раздел 1. Программирование на ЯВУ

Лабораторная 1.

Изучение процесса создания программы на ЯВУ. Среда программирования Delphi. Массивы. Работа с числовыми и строковыми массивами. Двухмерные массивы (матрицы). Подпрограммы. Создание подпрограмм обработки массивов (4 часа).

Лабораторная 2.

Неоднородные структуры данных. Тип запись в языках программирования. Работа с записями в языке Object Pascal в Delphi. Файлы. Запись информации в файлы на языке Object Pascal в Delphi (4 часа).

Семестр 2

Раздел 2. Программирование на ЯВУ (2 сем.)

Лабораторная 3.

Динамические структуры данных. Класс как динамическая структура данных. Создание классов в Delphi (4 часа).

Лабораторная 4.

Динамические структуры данных. Наследование элементов класса. Линейные списки, очереди, стеки. Стандартный класс, реализующий линейный список в Delphi (4 часа).

Семестр 3

Раздел 3. Программирование на ЯВУ (3 сем.)

Лабораторная 5.

Программирование свойств и обработчиков событий класса. Создание графического класса в Delphi (4 часа).

Лабораторная 6.

Создание класса визуального графического элемента управления. Создание компонента в Delphi. Создание пакета компонент (4 часа).

Лабораторная 7.

Изучение процесса программирования на языке C#. Программирование на языке C# в среде VisualStudio. Создание программы с графическим интерфейсом на языке C# с использованием объектов WindowsForms (4 часа).

Лабораторная 8.

Изучение процесса программирования на языке Java. Среда программирования NetBeans. Создание программы с графическим интерфейсом на языке Java (4 часа).

4.2.2.4. Перечень тем и учебно-методическое обеспечение самостоятельной работы

Перечень тем, вынесенных на самостоятельное изучение:

1. Языки программирования Object Pascal и C++.
2. Рекурсивные определения и алгоритмы. Программирование рекурсивных алгоритмов.
3. Модульное программирование. Модули. Модульные программы.
4. Регулярные типы данных.
5. Алгоритмы сортировки массивов.
6. Динамические переменные. Динамические структуры данных. Динамические массивы.
7. Стеки. Очереди. Линейные списки: основные виды и способы реализации; линейный список как абстрактный тип данных.
8. Концепция объектно-ориентированного программирования.
9. Способы воспроизведения звука.
10. Сканирование иерархической структуры директорий по рекурсивному алгоритму.
11. Среда программирования Delphi.
12. Знакомство со средой визуального программирования C++ Builder.
13. Программирование свойств и обработчиков событий класса. Создание графического класса в Delphi.
14. Программирование COM-объектов. Создание программы передачи данных и генерации документов Microsoft Excel.
15. Создание программы передачи данных и генерации документов Microsoft Word.

Для самостоятельной работы используются методические указания по освоению дисциплины и издания из списка приведенной ниже основной и дополнительной литературы.

4.2.2.5. Перечень тем контрольных работ, рефератов, ТР, РГР, РПР

1. Контрольная работа (1 семестр) - Разработка блок-схемы алгоритма и программы обработки массивов на языках высокого уровня Delphi и C++ (по вариантам).
2. Контрольная работа (3 семестр) - Разработка программы обработки массивов на объектно-ориентированных языках программирования C# и Java (по вариантам).

4.2.2.6. Примерный перечень тем курсовых работ (проектов)

1. Приложения ОС Windows. Технические расчеты.
2. Приложения ОС Windows. Экранные компьютерные игры.
3. Приложения ОС Windows. Логические компьютерные игры.
4. Приложения ОС Windows. Организация документации.
5. Приложения ОС Windows. Текстовые/графические редакторы.

5. Образовательные технологии

В процессе изучения дисциплины применяется контактная технология преподавания (за исключением самостоятельно изучаемых студентами вопросов). При проведении лабораторных работ применяется имитационный или симуляционный подход. Шаги решения задач студентам демонстрируются при помощи мультимедийной техники. В дальнейшем студенты самостоятельно решают аналогичные задания.

6. Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины.

Фонды оценочных материалов (средств) приведены в приложении.

7. Учебно-методическое и информационное обеспечение дисциплины.

7.1. Основная учебно-методическая литература по дисциплине

1. Кучунова, Е. В. Программирование. Процедурное программирование : учебное пособие / Е. В. Кучунова, Б. В. Олейников, О. М. Чередниченко. — Красноярск : Сибирский федеральный университет, 2016. — 92 с. — ISBN 978-5-7638-3555-7. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: http://www.iprbookshop.ru/84094.html - http://www.iprbookshop.ru/84094.html

2. Фарафонов, А. С. Программирование на языке высокого уровня : методические указания к проведению лабораторных работ по курсу «Программирование» / А. С. Фарафонов. — Липецк : Липецкий государственный технический университет, ЭБС АСВ, 2013. — 32 с. — ISBN 2227-8397. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: http://www.iprbookshop.ru/22912.html - http://www.iprbookshop.ru/22912.html

3. Костюкова Н.И. Программирование на языке Си [Электронный ресурс] : методические рекомендации и задачи по программированию / Н.И. Костюкова. — Электрон. текстовые данные. — Новосибирск: Сибирское университетское издательство, 2017. — 160 с. — 978-5-379-02016-3. — Режим доступа: http://www.iprbookshop.ru/65289.html - http://www.iprbookshop.ru/65289.html

4. Санников, Е. В. Курс практического программирования в Delphi. Объектно-ориентированное программирование / Е. В. Санников. — Москва : СОЛОН-ПРЕСС, 2016. — 188 с. — ISBN 978-5-91359-122-7. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: http://www.iprbookshop.ru/90323.html - http://www.iprbookshop.ru/90323.html

5. Зоткин, С. П. Программирование на языке высокого уровня C/C++ : конспект лекций / С. П. Зоткин. — 3-е изд. — Москва : МИСИ-МГСУ, ЭБС АСВ, 2018. — 140 с. — ISBN 978-5-7264-1810-0. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: http://www.iprbookshop.ru/76390.html - http://www.iprbookshop.ru/76390.html

6. 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. — ISBN 2227-8397. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: http://www.iprbookshop.ru/90876.html - http://www.iprbookshop.ru/90876.html

7. Медведев, М. А. Программирование на СИ# : учебное пособие / М. А. Медведев, А. Н. Медведев ; под редакцией А. В. Присяжный. — Екатеринбург : Уральский федеральный университет, ЭБС АСВ, 2015. — 64 с. — ISBN 978-5-7996-1561-1. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: http://www.iprbookshop.ru/69667.html - http://www.iprbookshop.ru/69667.html

8. Гуськова, О. И. Объектно ориентированное программирование в Java : учебное пособие / О. И. Гуськова. — Москва : Московский педагогический государственный университет, 2018. — 240 с. — ISBN 978-5-4263-0648-6. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: http://www.iprbookshop.ru/97750.html - http://www.iprbookshop.ru/97750.html

7.2. Дополнительная учебно-методическая литература по дисциплине

1. Программирование на языке высокого уровня [Электронный ресурс] : методические указания и варианты заданий для студентов 1-го курса направления подготовки 09.03.01 Информатика и вычислительная техника / . — Электрон. текстовые данные. — М. : Московский государственный строительный университет, Ай Пи Эр Медиа, ЭБС АСВ, 2016. — 89 с. — 978-5-7264-1277-1. — Режим доступа: http://www.iprbookshop.ru/46060.html - http://www.iprbookshop.ru/46060.html

2. Андреева, Т. А. Программирование на языке Pascal : учебное пособие / Т. А. Андреева. — 3-е изд. — Москва : Интернет-Университет Информационных Технологий (ИНТУИТ), Ай Пи Ар Медиа, 2020. — 277 с. — ISBN 978-5-4497-0688-1. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: http://www.iprbookshop.ru/97576.html - http://www.iprbookshop.ru/97576.html

3. Поляков, А. Ю. Программирование : практикум / А. Ю. Поляков, А. Ю. Полякова, Е. Н. Перышкова. — Новосибирск : Сибирский государственный университет телекоммуникаций и информатики, 2015. — 55 с. — ISBN 2227-8397. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: http://www.iprbookshop.ru/55494.html - http://www.iprbookshop.ru/55494.html

4. Алексеев, Г. В. Разработка электронных учебных изданий на основе языка HTML : учебно-методическое пособие / Г. В. Алексеев, И. И. Бриденко. — 2-е изд. — Саратов : Вузовское образование, 2019. — 99 c. — ISBN 978-5-4487-0433-8. — Текст : электронный // Электронно-библиотечная система IPR BOOKS : [сайт]. — URL: http://www.iprbookshop.ru/79673.html - http://www.iprbookshop.ru/79673.html

7.3. Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине, включая перечень программного обеспечения и информационных справочных систем

В образовательном процессе используются информационные технологии, реализованные на основе информационно-образовательного портала института (www.mivlgu.ru/iop), и инфокоммуникационной сети института:

- предоставление учебно-методических материалов в электронном виде;
- взаимодействие участников образовательного процесса через локальную сеть института и Интернет;
- предоставление сведений о результатах учебной деятельности в электронном личном кабинете обучающегося.

Информационные справочные системы:

Электронная библиотечная система IPRbooks (http://www.iprbookshop.ru)

Электронная библиотека «ЭВРИКА» (http://www.mivlgu.ru/content/elektronnaya-biblioteka-«evrika»)

Электронная библиотека ВлГУ (e.lib.vlsu.ru)

Программное обеспечение:

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

7.4. Перечень ресурсов информационно-телекоммуникационной сети «Интернет», необходимых для освоения дисциплины

iprbookshop.ru
mivlgu.ru
mivlgu.ru/iop

8. Материально-техническое обеспечение дисциплины

Лаборатория программирования и лицензионного программного обеспечения

Компьютер Kraftway Credo KC 36 - 12 шт.; проектор NEC Projector VT595G; экран настенный; акустическая система.

9. Методические указания по освоению дисциплины

Для успешного освоения теоретического материала обучающийся: знакомится со списком рекомендуемой основной и дополнительной литературы; уточняет у преподавателя, каким дополнительным пособиям следует отдать предпочтение; ведет конспект лекций и прорабатывает лекционный материал, пользуясь как конспектом, так и учебными пособиями.

На практических занятиях пройденный теоретический материал подкрепляется решением задач по основным темам дисциплины. Занятия проводятся в компьютерном классе, используя специальное программное обеспечение. Каждой подгруппе обучающихся преподаватель выдает задачу, связанную с разработкой алгоритма, интерфейса программы и

программы на языке программирования, ее отладкой и тестированием. В конце занятия обучающие демонстрируют полученные результаты преподавателю и при необходимости делают работу над ошибками.

До выполнения лабораторных работ обучающийся изучает соответствующий раздел теории. Перед занятием студент знакомится с описанием заданий для выполнения работы, внимательно изучает содержание и порядок проведения лабораторной работы. Лабораторная работа проводятся в компьютерном классе. Обучающиеся выполняют индивидуальную задачу разработки алгоритма, интерфейса программы и программы на языке программирования, ее отладкой и тестированием, в соответствии с заданием на лабораторную работу. Полученные результаты исследований сводятся в отчет и защищаются по традиционной методике в классе на следующем лабораторном занятии. Необходимый теоретический материал, индивидуальное задание, шаги выполнения лабораторной работы и требование к отчету приведены в методических указаниях, размещенных на информационно-образовательном портале института.

Самостоятельная работа оказывает важное влияние на формирование личности будущего специалиста, она планируется обучающимся самостоятельно. Каждый обучающийся самостоятельно определяет режим своей работы и меру труда, затрачиваемого на овладение учебным содержанием дисциплины. Он выполняет внеаудиторную работу и изучение разделов, выносимых на самостоятельную работу, по личному индивидуальному плану, в зависимости от его подготовки, времени и других условий.

Курсовая работа выполняется в соответствии с методическими указаниями на курсовую работу. Обучающийся выбирает одну из указанных в перечне тем курсовых работ, исходя из своих интересов, наличия соответствующих литературных и иных источников. В ходе выполнения курсовой работы преподаватель проводит консультации обучающегося. На заключительном этапе обучающийся оформляет пояснительную записку к курсовой работе и выполняет ее защиту в присутствии комиссии из преподавателей кафедры.

Форма заключительного контроля при промежуточной аттестации – экзамен. Для проведения промежуточной аттестации по дисциплине разработаны фонд оценочных средств и балльно-рейтинговая система оценки учебной деятельности студентов. Оценка по дисциплине выставляется в информационной системе и носит интегрированный характер, учитывающий результаты оценивания участия студентов в аудиторных занятиях, качества и своевременности выполнения заданий в ходе изучения дисциплины и промежуточной аттестации.

Программа составлена в соответствии с требованиями ФГОС ВО по направлению *09.03.01 Информатика и вычислительная техника* и профилю подготовки *Вычислительные машины, комплексы, системы и сети*

Рабочую программу составил *к.т.н., преподаватель Бейлекчи Д.В.*_____

Программа рассмотрена и одобрена на заседании кафедры *ЭиВТ* протокол № 34 от 29.05.2019 года.

Заведующий кафедрой *ЭиВТ* _____ *Кропотов Ю.А.*

(Подпись)

Рабочая программа рассмотрена и одобрена на заседании учебно-методической комиссии факультета ФРЭКС

протокол № 9 от 31.05.2019 года.

Председатель комиссии ФРЭКС _____ Белов А.А.

(Подпись)

Лист актуализации рабочей программы дисциплины

Программа одобрена на 2020/2021 учебный год.

Протокол заседания кафедры № 24 от 27.05.2020 года.

Заведующий кафедрой_____ _____ Кропотов Ю.А.
(Подпись)

Программа одобрена на 2021/2022 учебный год.

Протокол заседания кафедры № 32 от 19.05.2021 года.

Заведующий кафедрой_____ _____ Белов А.А.
(Подпись)

Программа одобрена на 2022/2023 учебный год.

Протокол заседания кафедры № 34 от 11.05.2022 года.

Заведующий кафедрой_____ _____ Белов А.А.
(Подпись)

Приложение

Фонд оценочных материалов (средств) по дисциплине

Программирование

1. Оценочные материалы для проведения текущего контроля успеваемости по дисциплине

Составить блок-схему алгоритма решения задачи и разработать программу на языке высокого уровня реализующую алгоритм решения:

1. Определить количество целых чисел, входящих в массив только один раз
2. Подсчитать количество вхождений каждого числа в массив
3. Перенести в конец массива все четные элементы
4. Определить элемент, имеющий минимальное отклонение от среднего значения элементов массива
5. Расположить ненулевые элементы массива по убыванию
6. Расположить элементы массива по возрастанию абсолютных значений
7. Расположить элементы массива, на нечетных местах, в порядке возрастания, а на нечетных в порядке убывания
8. Найти сумму чисел, расположенных между максимальным и минимальным элементами массива
9. Вычислить наибольшее и наименьшее значения модуля разности между соседними элементами массива
10. В массиве найти первую группу повторяющихся элементов, вывести количество повторяющихся элементов в этой группе
11. Дана строка символов. Подсчитать количество слов и количество символов во всех словах, состоящих только из цифр
12. Дана строка символов. Вывести слово, содержащее наибольшее количество цифр и вывести число цифр в каждом слове
13. Дана строка символов, состоящая из символов "0"-"9". Рассматривая данную последовательность как запись числа в десятичной системе счисления, составить алгоритм и программу перевода его числовую форму
14. Дана строка символов, состоящая из символов "0"-"9", "A"-"F". Рассматривая данную последовательность как запись числа в шестнадцатеричной системе счисления, составить алгоритм и программу перевода его числовую форму
15. Дана строка символов, число n и символ d. Вывести все слова, содержащие n символов d
16. Дана строка символов и строки символов a и b. Заменить в строке в каждую группу символов a на b
17. В матрице поменять местами строку с max элементом и строку с min элементом
18. Найти наибольший элемент среди стоящих на главной и побочной диагоналях и поменять его местами с элементом, стоящим на пересечении этих диагоналей
19. В матрице найти сумму элементов строки, в которой расположен элемент с наименьшим значением
20. В каждой строке поменять местами наибольший и наименьший элементы
21. Найти сумму элементов расположенных выше главной и побочной диагонали

Вопросы для устного опроса (3 семестр):

1. Что такое свойство класса?
2. Что такое и как определяется тип доступа к свойству класса?
3. Как определяются методы чтения и записи свойств?
4. Что такое обработчик события класса?
5. Как описывается обработчик события класса?
6. Как обеспечивается вызов обработчика события класса?
7. Как обеспечивается работа со свойствами в разработанном классе?

8. Как обеспечивается работа с обработчиком событий в разработанном классе?
9. Как обеспечивается вывод изображения в разработанном классе?
10. Что такое и как описывается компонент?
11. Что такое и как создать пакет компонентов?
12. Какие бывают типы пакетов?
13. Что такое оконный элемент управления?
14. Что такое графический неоконный элемент управления?
15. В каких режимах может работать компонент?
16. Как установить пакет компонентов в RAD Studio?
17. Как описываются свойства компонента, предназначенные для изменения в инспекторе объектов?
18. Где и как обеспечивается сохранение значения свойств компонентов в процессе разработки приложения?
19. Как обеспечивается вызов обработчиков событий в компоненте?
20. Что такое элемент управления ActiveX?
21. Где может использоваться элемент управления ActiveX?
22. Как создать элемент управления ActiveX в Delphi?
23. Какие интерфейсы реализует элемент управления ActiveX?
24. Какие файлы проекта описывают элемент управления ActiveX?
25. Каким образом можно использовать элементы управления ActiveX в Microsoft Office?
26. Как средства программирования используются для управления элементами ActiveX в Microsoft Office?
27. Каким образом можно использовать элементы управления ActiveX в Web-документах??
28. Как HTML-теги используются для внедрения элементов ActiveX в Web-документы?
29. Как средства программирования используются для управления элементами ActiveX в Web-документах?
30. Что такое OLE Automation?
31. Как реализуется поддержка OLE Automation в Delphi?
32. Для чего нужен тип OleVariant?
33. Что такое контроллер автоматизации приложений Microsoft Office?
34. Что такое объектная модель Microsoft Office?
35. Как реализуется поддержка контроллеров автоматизации Microsoft Office в Delphi?
36. Каким образом можно использовать OLE-документы в приложениях?
37. Как реализуется поддержка OLE-объектов в Delphi?
38. Какие существуют OLE-документы (примеры)?
39. Как обеспечить управление OLE-объектами в приложении?
40. Что такое .NET?
41. Какие основные характеристики технологии .NET?
42. Что такое CLR, CIL, CTS и FCL в технологии .NET?
43. Какие основные характеристики среды программирования Visual Studio?
44. Какова структура проекта на языке C# в Visual Studio?
45. Какие основные характеристики языка C#?
46. Какие классы реализуют стандартные типы данных в языке C#?
47. Что такое Windows Forms?
48. Какие классы Windows Forms использовались в приложении?
49. Как осуществляется вывод простейшей графики на языке C#?
50. Как организуются классы в языке C#?
51. Какие элементы может содержать класс в языке C#?
52. Как описывается класс в языке C#?
53. Что такое и как описываются данные класса (синтаксис, пример)?

54. Что такое и как описываются методы класса (синтаксис, пример)?
55. Что такое и как описываются свойства класса (синтаксис, пример)?
56. Что такое и как описываются конструкторы и деструктор класса (синтаксис, пример)?
57. Что такое операции класса?
58. Что такое индексатор класса?
59. Что такое делегаты в языке C#?
60. Что такое обработчики событий класса (синтаксис, пример)?
61. Что такое библиотека классов?
62. Как создать библиотеку классов?
63. Как использовать библиотеку классов?
64. Что такое технология Java?
65. Какие основные характеристики технологии Java?
66. Какие существуют спецификации технологии Java?
67. Что такое JRE и JDK в технологии Java?
68. Какие основные характеристики языка программирования Java?
69. Какие классы реализуют стандартные типы данных в языке Java?
70. Какие существуют среды программирования для языка Java?
71. Какие основные характеристики среды программирования NetBeans IDE?
72. Какова структура проекта в NetBeans IDE?
73. Что такое Swing в Java?
74. Какие классы Swing и AWT использовались в приложении?
75. Как организуются классы в языке Java?
76. Какие элементы может содержать класс в языке Java?
77. Как описывается класс в языке Java?
78. Что такое и как описываются данные класса в языке Java?
79. Что такое и как описываются методы класса в языке Java?
80. Как реализуются в языке Java свойства класса?
81. Как реализуются в языке Java конструкторы и деструктор класса?
82. Как реализуются в языке Java обработчики событий класса?
83. Что такое пакет классов в языке Java?
84. Как создать пакет классов в языке Java?
85. Как использовать пакет классов в языке Java?

Общее распределение баллов текущего контроля по видам учебных работ для студентов

Рейтинг-контроль 1	Устный опрос, отчеты по практическим и лабораторным работам	10 баллов
Рейтинг-контроль 2	Устный опрос, контрольная работа, отчеты по практическим и лабораторным работам	15 баллов
Рейтинг-контроль 3	Устный опрос, отчеты по практическим и лабораторным работам	20 баллов
Посещение занятий студентом	Контроль посещаемости	15 баллов

Дополнительные баллы (бонусы)	нет	0
Выполнение семестрового плана самостоятельной работы	нет	0

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)

Примерные вопросы для промежуточной аттестации студентов на экзамене.

ОПК-3.3:

Блок 1 (Знать).

Языками высокого уровня являются:

– Delphi
– C++
– Basic
– Assembler

Средства записи программ, реализующие машинно-зависимые правила представления выполняемых вычислительной машиной команд, называются …

– языками низкого уровня
– языками высокого уровня
– языками аппаратного уровня
– языками командного уровня

Средства записи программ, реализующие машинно-независимые правила представления алгоритмов определенного класса, называются …

– языками высокого уровня
– языками низкого уровня
– языками алгоритмического уровня
– языками абстрактного уровня

Транслятор – это программа, осуществляющая перевод программ с определенного языка программирования …

– в эквивалентную ей программу на другом языке или эквивалентный исполняемый двоичный код
– в эквивалентный выполняемый двоичный код
– в эквивалентную ей программу на другом языке
– в форму, предназначенную для выполнения программы

Программа, осуществляющая перевод программы с определенного языка программирования в код, предназначенный для выполнения, называется …

– компилятором
– интерпретатором
– компоновщиком
– переводчиком

Программа, осуществляющая перевод программы с определенного языка программирования поэтапно (покомандно) и/или сразу же выполняющая команды программы, называется …

– интерпретатором
– компилятором
– трассировщиком
– переводчиком

Часть среды программирования, обеспечивающая пошаговое выполнение программы с контролем промежуточного состояния программы, называется …

– отладчиком

– трассировщиком
– компилятором
– интерпретатором

ОПК-2.2:
Блок 2 (Уметь).
Алгоритм, представленный в графический форме называется …
– блок-схемой алгоритма
– моделью алгоритма
– рисунком алгоритма
– графом алгоритма

ОПК-2.3:
Блок 3 (Владеть).В какой библиотеке описаны подпрограммы преобразования типов:
– windows
– SysUtils
– StdCtrls

ОПК-8:
Блок 1 (Знать).
Алгоритм обладает свойствами:
– определенность
– дискретность
– постоянность
– универсальность
Алгоритм обладает свойствами:
– массовость
– результативность
– постоянность
– безошибочность
Основные средства представления алгоритмов:
– графический, текстовый, табличный
– графический, текстовый
– графический, табличный
– графический, текстовый, модельный
В методе структурной алгоритмизации используются структурные фрагменты:
– композиция, альтернатива, итерация
– альтернатива, итерация
– композиция, альтернатива, следование
– итерация, альтернатива, ветвление
Виды разветвленных алгоритмов:
– двуальтернативный, многоальтернативный
– одноальтернативный, многоальтернативный
– двуальтернативный, циклический
– многоальтернативный, циклический
Процедура отличается от функции тем, что она …
– не возвращает значение под своим именем
– может возвращать значения через параметры
– может использоваться в выражениях
– не имеет параметров
В заголовке подпрограммы в языках C++ и Delphi указываются:
– имя подпрограммы
– формальные параметры подпрограммы
– фактические параметры подпрограммы

– локальные переменные подпрограммы

Параметры, указываемые при вызове подпрограммы, называются …

– фактическими параметрами
– формальными параметрами
– задаваемыми параметрами
– записываемыми параметрами

Параметры подпрограмме могут передаваться:

– по значению
– по ссылке
– по выражению
– по описанию

Рекурсивная подпрограмма – это подпрограмма …

– вызывающая саму себя
– создающая свою копию
– вызывающая свою копию
– вызывающая другие подпрограммы

Начальное значение локальных переменных подпрограммы в языках C++ и Delphi …

– неопределенно
– равно нулю
– равно случайному значению
– пустое

Правильный пример описания переменной текстового файлового типа в языке Delphi:

– var f:textfile;
– var f:file of text;
– var f:file(text);
– var f:stringfile;

Типизированный файловый тип в Delphi описывается следующей конструкцией:

– var <файловая_переменная>:file of <тип_элемента>;
– var <файловая_переменная>:<тип_элемента>;
– var <файловая_переменная>:file(<тип_элемента>);
– var <файловая_переменная>:file <тип_элемента>;

В языках программирования процедуры чтения/записи для файла используют …

– последовательный доступ к данным файла
– прямой доступ к данным файла
– блочный доступ к данным файла
– двоичный доступ к данным файла
– параллельный доступ к данным файла

В языках программирования с файловой переменной связанны:

– указатель позиции чтения/записи
– буфер чтения/записи
– имя файла
– формат файла
– размер файла

В языке Delphi встроенные процедуры BlockRead и BlockWrite обеспечивают чтение/запись из …

– нетипизированных файлов
– типизированных файлов
– блочных файлов
– текстовых файлов

Блок 2 (Уметь).

Цикл с параметром это разновидность цикла …

– с предусловием
– с постусловием

– со счетчиком
– с перебором

В языке Delphi символ языка while определяет ...
– оператор цикла с предусловием
– оператор выбора
– оператор цикла с постусловием
– оператор цикла с параметром
– оператор ожидания

В языке Delphi символ языка repeat определяет ...
– оператор цикла с постусловием
– оператор цикла с предусловием
– оператор цикла с параметром
– оператор безусловного цикла

В языке Delphi символ языка for определяет ...
– оператор цикла с параметром
– оператор цикла с предусловием
– оператор цикла с постусловием
– оператор цикла с перебором
– оператор выбора

В языке Delphi символ языка case определяет …
– оператор выбора
– оператор условия
– оператор цикла с параметром
– оператор цикла с предусловием

Основные виды циклов в структурном программировании …
– цикл предусловием и цикл постусловием
– цикл параметром и цикл постусловием
– цикл выбором и цикл предусловием
– цикл с параметром и цикл с выбором

Цикл с предусловием отличается от цикла с постусловие тем, что …
– цикл с предусловием может ни разу не выполниться, а с постусловием выполняется хотя бы один раз
– цикл с постусловием может ни разу не выполниться, а с предусловием выполняется хотя бы один раз
– цикл с предусловием всегда проверяет условие выхода, а с постусловием не всегда
– цикл с предусловием не всегда проверяет условие выхода, а с постусловием всегда

В языках C++ и Delphi оператор continue обеспечивает …
– переход на начало цикла
– выход из цикла
– выход из оператора выбора
– продолжение выполнения цикла

В языках C++ и Delphi оператор break обеспечивает …
– прерывание цикла
– переход на начало цикла
– выход из оператора условия
– продолжение выполнения цикла

В языках C++ и Delphi существуют операторы:
– цикл с предусловием
– цикл с параметром
– оператор условия
– цикл с выбором
– оператор перебора

Правильные примеры оператора условия в языке Delphi:
– if (a>b) then begin c:=1;b:=4;end else begin a:=1;b:=5;end;
– if (a>b) then begin c:=1;b:=4;end else a:=1;b:=5;
– if (a>b) then c:=1 else a:=1;
– if (a>b) then c:=1; else a:=1;
– if (a>b) then c:=1;b:=4 else a:=1;b:=5;
– if (a>b) then c:=1;b=4 else begin a:=1;b:=5;end;

Блок 3 (Владеть).
Значение выражения целого типа: 6 and 3 or 8 равно ...
– 10
– 15
– 3
– 8
– 6
Значение выражения: byte(not 3 and 6) равно …
– 4
– 3
– 0
– 252
Значение выражения: 3 and 7 xor 5 равно …
– 6
– 3
– 7
– 4
Результат выполнения операции 5 mod 2:
– 2.5
– 1
– 2
– 5
Результат выполнения операции 5 div 2:
– 2.5
– 1
– 2
– 5
Правильный пример оператора присваивания в Delphi:
– x:= a+7;
– x= a+7;
– x<- a+7;
– x<=a+7;
Тип данных shortint в языке Delphi является …
– целым со знаком
– целым беззнаковым
– вещественным со знаком
– коротким двоичным
Тип данных double в языках C++ и Delphi является …
– вещественным типом данных
– целым типом данных
– динамическим типом данных
– двойным типом данных
Тип данных float в языке C++ является …
– вещественным типом данных
– целым типом данных
– динамическим типом данных

– плавающим типом данных

В языке Delphi логическим типом является тип …

– boolean
– logical
– binary
– bool

В языке Delphi символьный тип WideChar используется для представления …

– Unicode-символов
– расширенных символов
– ASCII-символов
– Win-символов

В языке Delphi символьным типом для представления 8-битных символов используется

– AnsiChar
– WideChar
– Char8
– ShortChar

В языке Delphi конструкция type TMode=(mOne, mTwo, mThree); описывает …

– перечисляемый тип данных
– константный тип данных
– множественный тип данных
– именованный тип данных

Выражение 23..40 в языке Delphi является значением …

– интервального типа
– тип-отрезок
– целого типа
– ограниченного целого типа

Правильный пример описания массива в языке Delphi:

– var a:array [1..20] of byte;
– type a:array [1..20] of byte;
– var a:array [20] of integer;
– var a=array [1..23] of byte;

В языке Delphi к строковому типу данных применимы операции:

– слияния
– отношения
– переименования
– сложения

Правильные примеры описания переменных в языке Delphi:

– var a,b:integer;
– var a:byte=5; b:integer;
– var a=5,b=7:byte;
– var a,b:byte=7;

Какое значение будет иметь переменная d для заданных a=5, b=3, c=1:

```
if (a<b) and (b<c) then d:= c-a+b
else
   if (a>b) and (b>c) then d:= a-b+c
   else
      if (a>b) and (b<c) then d:= a-c+b;
```

Найти значение c при a=3,b=5,c=1:

```
while a<b do
  begin
    c:= c+a*b;
    a:= a+1;
```

```
    end;
```

Найти значение c при a=10, b=5:

```
repeat
   b:= b+1;
   a:= a-b;
   c:= a+b;
until a?0;
```

Найти значение c для заданных a=3,b=2,c=4,n=4:

```
for  i:=1 to n do    c:= c+a*b;
```

Найти значение S:

```
for i:=1 to 6 do
S:=0;
 begin
  A [i]:=i;
  S:=S+A [i];
 end;
```

Найти значение S:

```
for i:=1 to 4 do
 begin
  A [i]:=i;
  S:=S*A [i];
 end;
S:=S/i;
```

Найти значение S:

```
S: = 1;
 for i: =1 to 5 do
  begin
    A [i]:=i;
    S:=S*A [i];
   end;
```

Найти значение d при a=3:

```
var a, b, c: integer; d: real;
begin
   b: = a*a;
   c: = a+b;
   d: = c/a;
end.
```

Найти значение c при a=2, b=4, c=3:

```
while a<b do
 begin
    c:= c+a*b;
    a:= a+1;
 end;
```

Найти значение c при a=9, b=4:

```
repeat
   b:= b+1;
```

```
    a:= a-b;
    c:= a+b;
 until a?0;
```

Найти значение c для заданных a=2,b=4,c=3,n=3:

```
  for i:=1 to n do  c:= c+a*b;
```

Найти значение S:

```
for i:=1 to 5 do
 begin
  A [i]:=i;
  S:=S*A [i];
 end;
```

Какое значение будет иметь переменная d для заданных a=2, в=3, c=4:

```
if (a<b) and (b<c) then d:= c-a+b
else
   if (a>b) and (b>c) then d:= a-b+c
   else
      if (a>b) and (b<c) then d:= a-c+b;
```

Найти значение c при a=3,в=5,c=2:

```
while a<b do
 begin
    c:= c+a*b;
    a:= a+1;
 end;
```

Найти значение c при a=8, b=3:

```
 repeat
     b:= b+1;
     a:= a-b;
     c:= a+b;
 until a?0;
```

Найти значение c для заданных a=3,b=2,c=5,n=3:

```
for i:=1 to n do
  c:= c+a*b;
```

Найти значение S:

```
for i:=1 to 6 do
  begin
      A [i]:=I;
      S:=S+A [i];
end;
  S:=S/I;
```

Вопросы билетов к экзамену (3 семестр):

ОПК-3.3:

Блок 1 (Знать).

1. Понятие объектно-ориентированного визуального программирования.
3. Библиотеки элементов управления в системах программирования.
8. COM-интерфейс.

14. Элементы управления ActiveX.
17. Технология OLE Automation.
24. Характеристики языка программирования C#.
25. Характеристики платформы .NET Framework.
26. Среда исполнения CLR.
27. Библиотека классов .NET.
34. Виртуальная машина JavaVM.
35. Среда выполнения программ JRE. JAR-пакеты.
Блок 2 (Уметь).
22. Общие принципы создания контроллеров автоматизации (OLE Automation).
21. Способы программирования приложений Microsoft Office.
23. Способы программирования контроллеров автоматизации Microsoft Office.
Блок 3 (Владеть).
12. Методы и средства программирования технологии COM.
20. Программирование OLE-объектов.
15. Программирование ActiveX-объектов.
16. Создание ActiveX-форм.

ОПК-8:
Блок 1 (Знать).
2. Понятие объектно-ориентированного компонента.
6. Цели и задачи технологии COM.
7. Базовые понятия технологии COM.
9. Идентификатор GUID в технологии COM.
10. Интерфейс IUnknown.
18. Реализация OLE Automation в языках программирования.
19. Тип данных Variant в технологии OLE.
33. Характеристики языка программирования Java.
29. Синтаксис языка C#.
30. Типы данных языка C#.
32. Классы в C#.
37. Синтаксис языка Java.
38. Типы данных языка Java.
40. Классы в Java.
Блок 2 (Уметь).
4. Понятие компонента в RAD Studio.
5. Принципы создания компонентов. Пакеты компонентов.
31. Операторы и выражения языка C#.
39. Операторы и выражения языка Java.
Блок 3 (Владеть).
11. Реализация интерфейсов.
13. Язык описания COM-интерфейсов IDL.
28. Средства разработки на языке C#.
36. Средства разработки на языке Java.

Методические материалы, характеризующих процедуры оценивания

Кроме защит лабораторных работ для проведения контроля успеваемости и промежуточной аттестации студентов применяется устный опрос и контрольная работа.

Для подготовки к опросам, контрольным работам, а также при выполнении курсового проектирования, лабораторных и практических работ студентам рекомендуется пользоваться методическими указаниями по дисциплине, приведенными в рабочей программе и/или в списке литературы.

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:
ОПК-3.3

В технологиях Java и .NET сборка мусора – это …
- автоматическое удаление объектов при выходе их из области видимости
- автоматическое удаление неиспользуемых объектов при компиляции программы
- автоматическая выгрузка из памяти неиспользуемых модулей программы

- оптимизация исходного текста программы с целью удаления неиспользуемых программных элементов

В языках программирования класс является …
- типом данных
- переменной
- свойством
- объектом

Виды разветвленных алгоритмов:
-двуальтернативный, многоальтернативный
-одноальтернативный, многоальтернативный
-двуальтернативный, циклический
-многоальтернативный, циклический

В методе структурной алгоритмизации используются структурные фрагменты: композиция, альтернатива и … (итерация)

ОПК-8.1

В технологиях Java и .NET сборка мусора – это …
- автоматическое удаление объектов при выходе их из области видимости
- автоматическое удаление неиспользуемых объектов при компиляции программы
- автоматическая выгрузка из памяти неиспользуемых модулей программы
- оптимизация исходного текста программы с целью удаления неиспользуемых программных элементов

Концепция объектно-ориентированного программирования, реализующая переопределение алгоритмов родительских методов в классе-наследнике называется ...
- полиморфизм
- инкапсуляция
- наследование
- итерация

Значение выражения: byte(not 3 and 6) равно …

Значение выражения: 7 xor 3 and 5 равно …

ОПК-8.2

Параметры, указываемые при вызове подпрограммы, называются …
- фактическими параметрами
- формальными параметрами
- задаваемыми параметрами
- записываемыми параметрами

S:Параметры подпрограмме могут передаваться:
- по значению
- по ссылке
- по выражению
- по описанию

S:Интерфейс класса предназначен для …
- описания методов

- описания методов и полей
- описания полей
- описания объектов

В ООП возможность реализации наследуемых свойств или методов отличающимися способами в рамках множества абстракций называется … (ответ в именительном падеже строчными буквами) (полиморфизм)

S:В ООП метод, выполняющий дополнительные действия при создании объекта класса, называется … (ответ в именительном падеже строчными буквами) (конструктор)

ОПК-8.3

Программа, осуществляющая перевод программы с определенного языка программирования в код, предназначенный для выполнения, называется …
- компилятором
- интерпретатором
- компоновщиком
- переводчиком

Начальное значение локальных переменных подпрограммы в языках C++ и Delphi …
- неопределенно
- равно нулю
- равно случайному значению
- пустое

Основным языком программирования технологии .NET является …
- язык C#
- язык VisualBasic
- язык C++
- язык Python

В языке C++ символ языка for определяет оператор цикла с ... (параметром)

В языке Delphi символ языка repeat определяет оператор цикла с ... (постусловием)

В языке C++ символ языка switch определяет оператор … (выбора)

ОПК-8.4

Тип данных float в языке C++ является …
- вещественным типом данных
- целым типом данных
- динамическим типом данных
- плавающим типом данных

В языках C# и Java неявный параметр метода с именем this определяет …
- ссылку на объект класса, вызвавший данный метод
- ссылку на параметры данного метода
- ссылку на данный метод класса
- ссылку на результат, возвращаемый данным методом

В языке Delphi к строковому типу данных применимы операции:
- слияния

- отношения
- переименования
- сложения

Найти значение c при a=2, b=4, c=3:

```
while a < b do
 begin
    c := c + a * b;
    a := a + 1;
 end;
```

Найти значение c для заданных a=2,b=4,c=3,n=2:

```
for i := 0 to n do  c := c + a * b;
```

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=2876&cat=27230%2C94390&category=27618%2C94390&qbshowtext=0&recurse=0&recurse=1&showhidden=0

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