

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»
(МОСКОВСКИЙ ПОЛИТЕХ)
Электростальский институт (филиал)
Московского политехнического университета

УТВЕРЖДАЮ

Директор
Электростальского института (филиала)
Московского политехнического
университета



/О.Д. Филиппова/

03.06.2026

Рабочая программа дисциплины
«Программирование и основы алгоритмизации»

Направление подготовки
27.03.04 Управление в технических системах

Направленность образовательной программы
«Информационные технологии в управлении»
(набор 2026-2027 года)

Квалификация (степень) выпускника
Бакалавр

Форма обучения
очная, очно-заочная

Электросталь 2026

Рабочая программа дисциплины разработана в соответствии с:

1) Приказом Министерства науки и высшего образования Российской Федерации от 31 июля 2020 г. № 871, федеральный государственный образовательный стандарт высшего образования – бакалавриат по направлению подготовки 27.03.04 Управление в технических системах.

2) Профессиональным стандартом 40.178 Специалист в области проектирования АСУ ТП, утвержденный приказом Министерства труда и социальной защиты Российской Федерации от «13» марта 2017 г. №272н.

3) Федеральный закон Российской Федерации от 29.12.2012 № 273-ФЗ «Об образовании в Российской Федерации» (с изменениями и дополнениями);

4) Приказ Министерства образования и науки Российской Федерации от 05.04.2017 № 301 «Об утверждении Порядка организации и осуществления образовательной деятельности по образовательным программам высшего образования – программа бакалавриата, программам специалитета, программам магистратуры».

5) Учебным планом (очной, очно-заочной форм обучения) по направлению подготовки 27.03.04 Управление в технических системах.

Рабочая программа дисциплины включает в себя оценочные материалы для проведения текущего контроля успеваемости и промежуточной аттестации по дисциплине (фонд оценочных средств для проведения текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины).

Автор: Т.В. Михайлова, доцент, к.т.н., кафедры ПМИИ
(указать ФИО, ученую степень, ученое звание или должность)

Программа одобрена на заседании кафедры ПМИИ (протокол № 7 от 03.06.2026 г.).

Согласовано:

Заведующий кафедрой «Прикладная математика и информатика», к.т.н.



/М.В. Лунева/

Руководитель образовательной программы
доцент, к.э.н.



/Н.В. Зиновьева/

Оглавление

1	Цель и задачи освоения дисциплины.....	4
2	Место дисциплины в структуре ООП бакалавриата.....	4
3	Структура и содержание дисциплины.....	5
3.1	Виды учебной работы и трудоемкость.....	5
3.2	Содержание разделов дисциплины.....	5
3.3	Тематика практических и лабораторных занятий.....	6
4	Учебно-методическое и информационное обеспечение дисциплины.....	7
4.1	Основная литература.....	7
4.2	Дополнительная литература.....	7
4.3	Электронные образовательные ресурсы.....	8
4.4	Лицензионное и свободно распространяемое программное обеспечение.....	8
4.5	Современные профессиональные базы данных и информационные справочные системы.....	8
5	Материально-техническое обеспечение дисциплины.....	8
6	Методические рекомендации.....	9
6.1	Методические рекомендации для обучающихся по освоению дисциплины.....	9
6.2	Методические рекомендации для преподавателя по организации обучения.....	9
7	Фонды оценочных средств.....	10
7.1	Методы контроля и оценивания результатов обучения.....	10
7.2	Шкала и критерии оценивания результатов обучения.....	11
7.3	Оценочные средства по дисциплине.....	26

1 Цель и задачи освоения дисциплины

Дисциплина «Программирование и основы алгоритмизации» относится к числу учебных дисциплин, формирующих базовые знания по направлению 15.03.04 Автоматизация технологических процессов и производств.

Целью дисциплины «Программирование и основы алгоритмизации» является:

- обучение студентов основным понятиям, идеям и методам структурного программирования,
- обучение студентов основным понятиям объектно-ориентированного программирования,
- приобретение студентами умений и навыков правильного написания программ, используя методы и понятия структурного программирования;
- научить распознавать и применять основные приемы и алгоритмы в конкретных прикладных задачах;
- освоение студентами современной культуры программирования, необходимой для изучения смежных и последующих дисциплин на основе принципа последовательного и непрерывного образования.

Задачи дисциплины

- достижение основной цели обеспечивается соответствием содержания разделов и тем программы «Программирование и основы алгоритмизации» задачам подготовки и уровню современных требований, предъявляемых к бакалавру;
- системностью и последовательностью изложения разделов и тем на лекциях и практических занятиях;
- повышением эффективности традиционных и применением новых методов и форм активного обучения;
- качественным текущим и итоговым контролем.

2 Место дисциплины в структуре ООП бакалавриата

Учебная дисциплина «Программирование и основы алгоритмизации» относится к обязательной части дисциплин ООП по направлению подготовки 15.03.04 Автоматизация технологических процессов и производств.

Для изучения данной дисциплины необходимы следующие знания, умения и навыки, формируемые предшествующими дисциплинами: математика.

Дисциплина «Программирование и основы алгоритмизации» взаимосвязана со следующими дисциплинами и практиками ООП:

Методы анализа и синтеза систем,
Компьютерные технологии в автоматизации отрасли,
Теория автоматического управления,
Инжиниринг технических систем отрасли,
Проектная деятельность.

Перечень планируемых результатов обучения по дисциплине (модулю), соотнесенные с планируемыми результатами освоения образовательной программы

В результате освоения дисциплины (модуля) у обучающихся формируются следующие компетенции и должны быть достигнуты следующие результаты обучения как этап формирования соответствующих компетенций

В результате освоения образовательной программы обучающийся должен обладать	Индикатор достижения компетенции	Перечень планируемых результатов обучения по дисциплине
ОПК-14 Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения	ИОПК-14.1. Владеет базовыми знаниями формирования алгоритмов для практического применения ИОПК-14.2. Разрабатывает алгоритмы управления и программы, пригодные для практического применения в системах автоматизированного управления.	Знать: общие принципы построения алгоритмов, основные алгоритмические конструкции; понятие системы программирования; основные элементы процедурного языка программирования, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, кассы памяти; подпрограммы, составление библиотек программ; объектно-ориентированную модель программирования, понятие классов и объектов, их свойств и методов. Уметь: использовать современные информационные технологии, технику, прикладные программные средства при решении профессиональных задач; использовать языки программирования, строить логически правильные и эффективные программы; распознавать в конкретных прикладных задачах те или иные алгоритмы решения. Владеть: навыками использования современных информационных технологий в проектно-конструкторской деятельности; методами и приемами решения прикладных задач; методикой программирования этих задач.

3 Структура и содержание дисциплины

3.1 Виды учебной работы и трудоемкость

Общая трудоемкость дисциплины составляет 8 зачетных единиц, т.е. 288 часов (из них 144 часа – самостоятельная работа студентов очной формы обучения и 216 часов очно-заочной формы обучения).

Разделы дисциплины изучаются очная форма обучения:

в 2-м семестре: лекции – 36 часов, практические занятия – 18 часов, лабораторные работы – 18 часов, форма контроля – зачет;

в 3-м семестре: лекции – 36 часов, практические занятия – 18 часов, лабораторные работы – 18 часов, форма контроля – экзамен.

Разделы дисциплины изучаются очно-заочная форма обучения:

в 1-м семестре: лекции – 18 часов, практические занятия – 12 часов, лабораторные работы – 6 часов, форма контроля – зачет;

во 2-м семестре: лекции – 18 часов, практические занятия – 12 часов, лабораторные работы – 6 часов, форма контроля – экзамен.

Структура и содержание дисциплины «Программирование и основы алгоритмизации» по срокам и видам работы отражены в Приложении А.

3.2 Содержание разделов дисциплины

Тематика лекций

2 семестр очной и 1 семестр очно-заочной формы обучения

Тема 1. Основные понятия в информатике

Алгоритм и программа.

Основные конструкции в программировании.

Язык Паскаль и среда разработки.

Тема 2. Типы данных

Основные операции над вещественными и символьными типами данных.

Множества и операции над множествами.

Тема 3. Массивы

Массивы: одномерные и двумерные.

Классические задачи для одномерных

массивов. Классические задачи для двумерных

массивов. Графические задачи на Паскале.

3 семестр очной и 2 семестр очно-заочной формы обучения

Тема 4. Записи

Что такое записи?

Основные методы работы с записями.

Тема 5. Основы объектно-ориентированного программирования

Термины в объектно-ориентированном программировании.

Объекты и их свойства.

Объекты и их методы.

Тема 6. Язык Си

Введение в язык Си.

Типы данных в языке Си и

операции. Типовые задачи на

языке Си.

Графика в Си.

3.3 Тематика практических и лабораторных занятий

2 семестр очной и 1 семестр очно-заочной формы обучения

Ввод и вывод на экран основных типов данных.

Форматированный вывод на экран переменных вещественного и целого типа.

Конструкция ветвления if ..then.

Конструкция ветвления caseof.

Циклическая конструкция с предусловием (while).

Циклическая конструкция с постусловием (repeat... until).

Циклическая конструкция с параметром (for).

Основные арифметические операции над вещественными и целыми типами данных.

Основные операции над строковыми типами данных.

Множества и операции над ними.

Ввод и вывод одномерных массивов.

Основные приемы работы с одномерными массивами.

Поиск минимума/максимума в одномерном массиве.

Ввод и вывод двумерных массивов (матриц).

Основные операции над матрицами.

Поиск минимального/максимального элемента в матрице.

Методы сортировки массивов.

Элементарная графика (построение функций, простые рисунки).

3 семестр очной и 2 семестр очно-заочной формы обучения

Создание записей.

Сортировка записей.

Фильтрация записей.

Классические задачи с записями.

Создание классов объекта

Знакомство со свойствами объекта.

Создание методов объекта.
 Классические задачи с объектами. Ввод и вывод информации на Си. Типы данных в Си.
 Работа с одномерными массивами в Си.
 Работа с двумерными массивами в Си.
 Методы сортировок в Си.
 Работа со строками.
 Функции в языке Си (стандартные и собственные).
 Графика в Си.

Самостоятельная работа обучающегося

Кол. час	Темы, разделы, вынесенные на самостоятельную подготовку, вопросы к практическим занятиям; тематика контрольных работ и др.
144/258	Самостоятельное изучение отдельных тем курса: Семестр 3 1 Циклическая конструкция с предусловием и постусловием. 2 Циклическая конструкция for. 3 Форматированный вывод данных на экран. 4 Работа со множествами. 5 Ввод и вывод массивов. 6 Работа с символьными переменными и символьными массивами. 7 Методы сортировок массивов. Семестр 4 1 Основные понятия объектно-ориентированного программирования. 2 Свойства и методы объектов. 3 Описание классов объектов на языке Си. 4 Основные операции над различными типами данных в Си. 5 Работа с символьными переменными и символьными массивами. 6 Методы сортировок массивов на языке Си

4 Учебно-методическое и информационное обеспечение дисциплины

4.1 Основная литература

1. Макарова Н.В., Волков В.Б. Информатика: Учебник для бакалавров. – СПб.: Питер, 2011. – 576с.
2. С.В. Симонович Информатика. Базовый курс. Учебник для вузов – С.Пб.: Питер 2009. – 640с.
3. Зыков С.В. Введение в теорию программирования. Функциональный подход: Учебное пособие. М.: Национальный Открытый Университет «ИНТУИТ», 2016. – 153с.
https://biblioclub.ru/index.php?page=book_red&id=457415&sr=1

4.2 Дополнительная литература

- 1 В.А. Острейковский. Информатика. – М: Высшая школа, 2007. – 511 с.
- 2 Фаронов В.В. Турбо Паскаль 7.0. Практика программирования: Учебное пособие. – М.: КноРус, 2009. – 416с.

3 Ездаков А.Л. Функциональное и логическое программирование: Учебное пособие – М.: БИНОМ, 2011. – 119с.

4.3 Электронные образовательные ресурсы

Учебно-методические материалы в электронном виде, представлены на сайте Электронная библиотека НТБ Московского политехнического университета <http://lib.mospolitech.ru>; <http://elib-mgu.ru>

ЭБС «Университетская книга онлайн» [http:// biblioclub.ru/](http://biblioclub.ru/);

ЭБС « Лань» www.e.lanbook.com;

Национальная электронная библиотека <https://rusneb.ru/>;

Электронно-библиотечная система и образовательная платформа «Юрайт» <https://urait.ru/>.

4.4 Лицензионное и свободно распространяемое программное обеспечение

– Операционная система Windows 7 DreamSpark № 9d0e9d49-31d1-494a-b303-612508131616

– Офисные приложения, Microsoft Office 2013 (или ниже) – Microsoft Open License. Лицензия № 61984042

– Microsoft Project 2013 Standart 32- bit/x64 Russian.

– Антивирусное ПО Avast (бесплатная версия)

– Turbo C++ (свободная лицензия)

– Turbo Pascal 7.1 (свободная лицензия)

– VBA 7.0 (свободная лицензия)

– Delphi7.0 (бесплатно для образовательных целей)

– Linux Ubuntu (свободная лицензия)

– Arduino 1.6.5 (свободная лицензия)

– 1С: Предприятие 8.2 (версия для обучения)

– AnyLogic (версия пакета имитационного моделирования бесплатно для образовательных целей)

– Forex Optimizer, LiteUpdateDevelop – программное обеспечение для работы на учебном сегменте рынка Форекс (свободная лицензия)

– XAMPP (свободная лицензия)

– MySQL (свободная лицензия).

4.5 Современные профессиональные базы данных и информационные справочные системы

Полезные учебно-методические и информационные материалы представлены на сайтах: <http://pas1.ru/> – ЯзыкPascal. Программирование для начинающих Язык Pascal. Программирование для начинающих

http://life-prog.ru/view_cat.php?cat=1 - Язык программирования Паскаль(TurboPascal). Обучающие уроки

<http://bourabai.kz/alg/pascal/index.html> - Язык программирования Паскаль

<http://info-comp.ru/programmirovanie/67-turbopascal-.html> - Основы среды программирования Турбо Паскаль

<http://www.cyberforum.ru/pascalabc/thread538933.html> - Pascal ABC - Графика в языках программирования

<http://danfa.ru/viewtopic.php?f=35&t=213> - Уроки Pascal ABC - Введение

Изучение дисциплины «Программирование и основы алгоритмизации» предполагает использование мультимедийных учебных аудиторий или аудиторий, оснащенных видеопроектором и компьютером.

5 Материально-техническое обеспечение дисциплины

Наименование специальных помещений и помещений для самостоятельной работы	Оснащенность специальных помещений и помещений для самостоятельной работы
Учебная аудитория лекционного типа № 501, учебно-лабораторный корпус, 144000, Московская область, г. Электросталь, ул. Первомайская, д.7	Комплект мебели, переносной мультимедийный комплекс (проекционный экран, проектор, ноутбук)

Компьютерные классы № 305, 306, учебно-лабораторный корпус, 144000, Московская область, г. Электросталь, ул. Первомайская, д.7	Комплект мебели, компьютеры, проектор.
--	--

6 Методические рекомендации

6.1 Методические рекомендации для обучающихся по освоению дисциплины

Время, отводимое на самостоятельную работу должно затрачиваться студентами для изучения лекционного материала, выполнение практических задач и подготовку к лабораторным работам (при их наличии). Самостоятельная работа студентов в ходе семестра является важной составной частью учебного процесса и необходима для закрепления и углубления знаний, полученных в период сессии на лекциях, практических и интерактивных занятиях, а также для индивидуального изучения дисциплины в соответствии с программой и рекомендованной литературой.

Лекции и частично практические занятия базируются на литературных источниках, указанных в основном и дополнительном списках литературы, приведенных в рабочей программе. Более детальные и подробные рекомендации по использованию в самостоятельной работе литературных источников, а также программного обеспечения, даются на занятиях преподавателем. На этих же занятиях преподаватель передает студентам интернет-ссылки или на флэшке видеоматериалы по лабораторным работам.

Особое внимание следует обратить на определение основных понятий курса. Студент должен подробно разбирать примеры, которые поясняют такие определения, и уметь строить аналогичные примеры самостоятельно. Нужно добиваться точного представления о том, что изучаешь. Полезно составлять опорные конспекты.

В конце рабочей программы есть контрольные вопросы, по которым студент имеет возможность самоконтроля выполненной работы.

В ряде дисциплин предусмотрены домашние задания, которые выполняются студентами в указанные преподавателем периоды времени (семестра). При этом студентом используются возможности представления выполненной работы в виде реферата, презентации или эссе.

При подготовке к контрольным мероприятиям, в том числе, защите курсовых проектов (работ), экзаменам и зачетам студент пользуется конспектами лекций, примерами выполнения практических расчетов, видеоматериалами и заполненными на лабораторных работах бланками по их выполнению. Преподавателем контроль качества самостоятельной работы может осуществляться с помощью устного опроса на лекциях или практических занятиях, тестирования, проведения коллоквиума, защиты презентации, эссе или рефератов, проверки письменных контрольных работ и реферативных обзоров.

Перед контрольными мероприятиями преподаватель выдает примерные вопросы, основная доля которых представлена в рабочей программе.

Методические указания для обучающихся по организации самостоятельной работы

Самостоятельная работа обучающихся направлена на самостоятельное изучение отдельной темы учебной дисциплины. Самостоятельная работа является обязательной для каждого обучающегося, ее объем определяется учебным планом. При самостоятельной работе студент взаимодействует с рекомендованными материалами при участии преподавателя в виде консультаций. Для выполнения самостоятельной работы предусмотрено методическое обеспечение. Электронно-библиотечной система (электронная библиотека) обеспечивает возможность индивидуального доступа каждого обучающегося из любой точки, в которой имеется доступ к сети Интернет.

6.2 Методические рекомендации для преподавателя по организации обучения

1. Изучив глубоко содержание учебной дисциплины, целесообразно разработать матрицу наиболее предпочтительных методов обучения и форм самостоятельной работы студентов, адекватных видам лекционных и семинарских занятий.
2. Необходимо предусмотреть развитие форм самостоятельной работы, выводя студентов к завершению изучения учебной дисциплины на её высший уровень.
3. Организуя самостоятельную работу, необходимо постоянно обучать студентов методам такой работы.

4. Вузовская лекция - главное звено дидактического цикла обучения. Её цель – формирование у студентов ориентировочной основы для последующего усвоения материала методом самостоятельной работы. Содержание лекции должно отвечать следующим дидактическим требованиям:

- изложение материала от простого к сложному, от известного к неизвестному;
- логичность, четкость и ясность в изложении материала;
- возможность проблемного изложения, дискуссии, диалога с целью активизации деятельности студентов;
- опора смысловой части лекции на подлинные факты, события, явления, статистические данные;
- тесная связь теоретических положений и выводов с практикой и будущей профессиональной деятельностью студентов.

Преподаватель, читающий лекционные курсы в вузе, должен знать существующие в педагогической науке и используемые на практике варианты лекций, их дидактические и воспитывающие возможности, а также их методическое место в структуре процесса обучения.

5. При изложении материала важно помнить, что почти половина информации на лекции передается через интонацию. В профессиональном общении исходить из того, что восприятие лекций студентами очно-заочной формы обучения существенно отличается по готовности и умению от восприятия студентами очной формы.

6. При проведении аттестации студентов важно всегда помнить, что систематичность, объективность, аргументированность – главные принципы, на которых основаны контроль и оценка знаний студентов. Проверка, контроль и оценка знаний студента, требуют учета его индивидуального стиля в осуществлении учебной деятельности. Знание критериев оценки знаний обязательно для преподавателя и студента.

Особенности реализации дисциплины для инвалидов и лиц с ограниченными возможностями здоровья

Обучение по дисциплине «Программирование и основы алгоритмизации» инвалидов и лиц с ограниченными возможностями здоровья (далее ОВЗ) осуществляется преподавателем с учетом особенностей психофизического развития, индивидуальных возможностей и состояния здоровья таких обучающихся.

Обучающиеся с ограниченными возможностями здоровья обеспечены электронными образовательными ресурсами в формах, адаптированных к ограничениям их здоровья.

По дисциплине «Программирование и основы алгоритмизации» обучение инвалидов и лиц с ограниченными возможностями здоровья может осуществляться как в аудитории, так и дистанционно с использованием возможностей электронной образовательной среды (образовательного портала) и электронной почты.

7 Фонды оценочных средств

7.1 Методы контроля и оценивания результатов обучения

Методика преподавания дисциплины «Программирование и основы алгоритмизации» и реализация компетентного подхода в изложении и восприятии предусматривает использование следующих активных и интерактивных форм проведения групповых, индивидуальных, аудиторных занятий в сочетании с внеаудиторной работой с целью формирования и развития профессиональных навыков, обучающихся:

- подготовка, обсуждение и написание программ;
- проведение интерактивных занятий по процедуре написания программного кода и поиска ошибок в нем;
- использование интерактивных форм текущего контроля в форме аудиторного тестирования.

Оценочные средства для текущего контроля успеваемости, промежуточной аттестации по итогам освоения дисциплины и учебно-методическое обеспечение самостоятельной работы студентов

В процессе обучения используются следующие оценочные формы самостоятельной

работы студентов, оценочные средства текущего контроля успеваемости и промежуточных аттестаций:

- вопросы для устного опроса,
- вопросы к экзамену,
- вопросы к зачету,
- контрольная работа,
- фонд тестовых заданий.

Образцы контрольных вопросов и заданий для проведения текущего и промежуточного контроля приведены в Приложении Б.

Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы

В результате освоения дисциплины (модуля) формируются следующие компетенции:

Код компетенции	Содержание компетенции
ОПК-14	Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения

В процессе освоения образовательной программы данные компетенции, в том числе их отдельные компоненты, формируются поэтапно в ходе освоения обучающимися дисциплин (модулей), практик в соответствии с учебным планом и календарным графиком учебного процесса.

7.2 Шкала и критерии оценивания результатов обучения

Показателем оценивания компетенций на различных этапах их формирования является достижение обучающимися планируемых результатов обучения по дисциплине (Таблица 2).

Таблица 2

Показатель	Критерии оценивания			
	2	3	4	5
ОПК-14 – Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения				
Знать: общие принципы построения алгоритмов, основные алгоритмические конструкции; понятие системы программирования; основные элементы процедурного языка программирования, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, кассы памяти; подпрограммы, составление библиотек программ; объектно-ориентированную модель программирования, понятие классов и объектов, их свойств и методов.	Обучающийся демонстрирует полное отсутствие или недостаточное соответствие знаний общих принципов построения алгоритмов, основных алгоритмических конструкций.	Обучающийся демонстрирует неполное соответствие знаний понятия системы программирования, основных элементов процедурного языка программирования, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, кассы памяти. Допускаются значительные ошибки, проявляется недостаточность знаний подпрограмм, составления библиотек программ, обучающийся испытывает значительные затруднения при оперировании знаниями при их переносе на новые ситуации.	Обучающийся демонстрирует частичное соответствие знаний объектно-ориентированной модели программирования, понятие классов и объектов, их свойств и методов, основных элементов процедурного языка программирования, структуры программы, операторы и операции, управляющие структуры, структуры данных, файлы, кассы памяти, но допускаются незначительные ошибки, неточности, затруднения при аналитических операциях.	Обучающийся демонстрирует полное соответствие необходимых знаний, свободно оперирует приобретенными знаниями.
Уметь: использовать современные информационные технологии, технику, прикладные программные средства при решении профессиональных задач; использовать языки программирования, строить логически правильные и эффективные программы; распознавать в конкретных прикладных задачах те или иные алгоритмы решения.	Обучающийся не умеет или в недостаточной степени умеет использовать языки программирования, строить логически правильные и эффективные программы; распознавать в конкретных прикладных задачах те или иные алгоритмы решения	Обучающийся демонстрирует неполное соответствие умений использовать языки программирования, строить логически правильные и эффективные программы; распознавать в конкретных прикладных задачах те или иные алгоритмы решения. Допускаются значительные ошибки, проявляется недостаточность умений использовать языки программирования, строить логически правильные и эффективные программы, обучающийся испытывает значительные затруднения при оперировании умениями при их переносе на новые ситуации.	Обучающийся демонстрирует частичное соответствие умений распознавать в конкретных прикладных (технических, экономических, социальных и т.п.) задачах математические модели из соответствующих разделов курса и проводить анализ этих моделей на основе изученных методов. Умения освоены, но допускаются незначительные ошибки, неточности, затруднения при аналитических операциях, переносе умений на новые, нестандартные ситуации.	Обучающийся демонстрирует полное соответствие умений, предъявляемых к данной компетенции. Свободно оперирует приобретенными умениями, применяет их в ситуациях повышенной сложности.
Владеть: навыками использования современных информационных технологий в проектно-конструкторской деятельности;	Обучающийся не владеет или в недостаточной степени владеет методами и	Обучающийся владеет методами и приемами решения прикладных задач в неполном объеме, допускаются значительные ошибки, проявляется недо-	Обучающийся частично владеет методами и приемами решения прикладных задач, методикой моделирования процессов, навыки	Обучающийся в полном объеме владеет навыками, предъявляемые к данной компе-

методами и приемами решения прикладных задач; методикой программирования этих задач.	приемами решения прикладных задач, методикой программирования этих задач.	статочность владения методикой моделирования процессов. Обучающийся испытывает значительные затруднения при применении навыков в новых ситуациях.	освоены, но допускаются незначительные ошибки, неточности, затруднения при аналитических операциях, переносе умений на новые, нестандартные ситуации.	тенции свободно применяет полученные навыки в ситуациях повышенной сложности.
--	---	---	---	---

Шкалы оценивания результатов промежуточной аттестации и их описание:

Форма промежуточной аттестации: экзамен

Промежуточная аттестация обучающихся в форме экзамена проводится по результатам выполнения всех видов учебной работы, предусмотренных учебным планом по данной дисциплине (модулю), при этом учитываются результаты текущего контроля успеваемости в течение семестра. Оценка степени достижения обучающимися планируемых результатов обучения по дисциплине (модулю) проводится преподавателем, ведущим занятия по дисциплине (модулю) методом экспертной оценки. По итогам промежуточной аттестации по дисциплине (модулю) выставляется оценка «отлично», «хорошо», «удовлетворительно» или «неудовлетворительно».

К промежуточной аттестации допускаются только студенты, выполнившие все виды учебной работы, предусмотренные рабочей программой по дисциплине «Программирование и основы алгоритмизации» (прошли промежуточный контроль, выполнили практические работы).

Шкала оценивания	Описание
Отлично	Выполнены все виды учебной работы, предусмотренные учебным планом. Студент демонстрирует соответствие знаний, умений, навыков приведенным в таблицах показателей, оперирует приобретенными знаниями, умениями, навыками, применяет их в ситуациях повышенной сложности. При этом могут быть допущены незначительные ошибки, неточности, затруднения при аналитических операциях, переносе знаний и умений на новые, нестандартные ситуации.
Хорошо	Выполнены все виды учебной работы, предусмотренные учебным планом. Студент демонстрирует неполное, правильное соответствие знаний, умений, навыков приведенным в таблицах показателей, либо если при этом были допущены 2-3 несущественные ошибки.
Удовлетворительно	Выполнены все виды учебной работы, предусмотренные учебным планом. Студент демонстрирует соответствие знаний, в котором освещена основная, наиболее важная часть материала, но при этом допущена одна значительная ошибка или неточность.
Неудовлетворительно	Не выполнен один или более видов учебной работы, предусмотренных учебным планом. Студент демонстрирует неполное соответствие знаний, умений, навыков приведенным в таблицах показателей, допускаются значительные ошибки, проявляется отсутствие знаний, умений, навыков по ряду показателей, студент испытывает значительные затруднения при оперировании знаниями и умениями при их переносе на новые ситуации.

Форма промежуточной аттестации: зачет.

Промежуточная аттестация обучающихся в форме зачёта проводится по результатам выполнения всех видов учебной работы, предусмотренных учебным планом по данной дисциплине (модулю), при этом учитываются результаты текущего контроля успеваемости в течение семестра. Оценка степени достижения обучающимися планируемых результатов обучения по дисциплине (модулю) проводится преподавателем, ведущим занятия по дисциплине (модулю) методом экспертной оценки. По итогам промежуточной аттестации по дисциплине (модулю) выставляется оценка «зачтено» или «не зачтено».

К промежуточной аттестации допускаются только студенты, выполнившие все виды учебной работы, предусмотренные рабочей программой.

Шкала оценивания	Описание
Зачтено	Выполнены все виды учебной работы, предусмотренные учебным планом. Студент демонстрирует соответствие знаний, умений, навыков приведенным в таблицах показателей, оперирует приобретенными знаниями, умениями, навыками, применяет их в ситуациях повышенной сложности. При этом могут быть допущены незначительные ошибки, неточности, затруднения при аналитических операциях, переносе знаний и умений на новые, нестандартные ситуации.
Не зачтено	Не выполнен один или более видов учебной работы, предусмотренных учебным планом. Студент демонстрирует неполное соответствие знаний,

	умений, навыков приведенным в таблицах показателей, допускаются значительные ошибки, проявляется отсутствие знаний, умений, навыков по ряду показателей, студент испытывает значительные затруднения при оперировании знаниями и умениями при их переносе на новые ситуации.
--	--

Фонды оценочных средств представлены в Приложении Б к рабочей программе.

n/n	Раздел	Семестр	Виды учебной работы, включая самостоятельную работу студентов, и трудоемкость в часах					Виды самостоятельной работы студентов *						Формы аттестации	
			Л	П/С	Лаб	СРС	КСР	ДС	УО	РЗЗ	Р	К/р	Т	Э	З
1.1	Основные понятия в информатике	2	12	4	-	14			+			+	+		
1.2	Типы данных		12	6	8	20			+			+	+		
1.3	Массивы		12	8	18	20			+			+	+		
	<i>Форма аттестации</i>	2							1			1	1		3
	Всего часов по дисциплине во втором семестре		36	18	18	54									
2.1	Записи	3	12	4	-	30			+			+	+		
2.2	Основы объектно-ориентированного программирования		12	8	8	30			+			+	+		
2.3	Язык Си		12	6	10	30			+			+	+		
	<i>Форма аттестации</i>	3							1			1	1	Э	
	Всего часов по дисциплине в третьем семестре		36	18	18	90									
	Всего по дисциплине		72	36	36	144									

Очно-заочная форма обучения

n/n	Раздел	Семестр	Виды учебной работы, включая самостоятельную работу студентов, и трудоемкость в часах					Виды самостоятельной работы студентов *						Формы аттестации	
			Л	П/С	Лаб	СРС	КСР	ДС	УО	РЗЗ	Р	К/р	Т	Э	З
1.1	Основные понятия в информатике	1	6			36			+			+	+		
1.2	Типы данных		6	6	2	36			+			+	+		
1.3	Массивы		6	6	4	36			+			+	+		
	<i>Форма аттестации</i>	1							1			1	1	Э	
	Всего часов по дисциплине в первом семестре		18	12	6	108									
2.1	Записи	2	4	2		36			+			+	+		
2.2	Основы объектно-ориентированного программирования		6	4	2	36			+			+	+		
2.3	Язык Си		8	6	4	36			+			+	+		
	<i>Форма аттестации</i>	2							1			1	1	Э	
	Всего часов по дисциплине во втором семестре		18	12	6	108									
	Всего по дисциплине		36	24	12	216									

* – Сокращения форм оценочных средств см. в Приложении В к РП.

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»
(МОСКОВСКИЙ ПОЛИТЕХ)
Электростальский институт (филиал)
Московского политехнического университета

Направление подготовки
27.03.04 Управление в технических системах

Направленность образовательной программы
«Информационные технологии в управлении»

Форма обучения: очная, очно-

заочная
Виды профессиональной деятельности:
проектно-конструкторская.

Кафедра «Прикладная математика и информатика»

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ
ПО ДИСЦИПЛИНЕ
«Программирование и основы алгоритмизации»

Состав: 1) Паспорт фонда оценочных средств
2) Описание оценочных средств:
вопросы для устного опроса,
вопросы к экзамену,
вопросы к зачёту,
контрольная работа,
фонд тестовых
заданий.

Составитель:

Т.В. Михайлова

Электросталь 2026

Паспорт
фонда оценочных средств по дисциплине
«Программирование и основы алгоритмизации»

Направление подготовки
27.03.04 Управление в технических системах

Направленность образовательной программы
«Информационные технологии в управлении»

Уровень
бакалавриата

Форма обучения
очная, очно-заочная

Контролируемые разделы (темы) дисциплины	Код контролируемой компетенции (или ее части)	Наименование оценочного средства
1 Основные понятия в информатике	ОПК-14	УО, Т, К/Р
2 Типы данных	ОПК-14	УО, Т, К/Р
3 Массивы	ОПК-14	УО, Т, К/Р
Промежуточная аттестация		Экзамен
4 Записи	ОПК-14	УО, Т, К/Р
5 Основы объектно-ориентированного программирования	ОПК-14	УО, Т, К/Р
6 Язык Си	ОПК-14	УО, Т, К/Р
Промежуточная аттестация		Зачёт
Промежуточная аттестация		Экзамен

Показатель уровня сформированности компетенций

Программирование и основы алгоритмизации				
ФГОС ВО 27.03.04 Управление в технических системах				
КОМПЕТЕНЦИИ	Перечень компонентов	Технология формирования компетенций	Форма оценочного средства	Степени уровней освоения компетенций
ИНДЕКС ФОРМУЛИРОВКА				
ОПК-14 Способен разрабатывать алгоритмы и компьютерные программы, пригодные для практического применения	Знать: общие принципы построения алгоритмов, основные алгоритмические конструкции; понятие системы программирования; основные элементы процедурного языка программирования, структуру программы, операторы и операции, управляющие структуры, структуры данных, файлы, кассы памяти; подпрограммы, составление библиотек программ; объектно-ориентированную модель программирования, понятие классов и объектов, их свойств и методов. Уметь: использовать современные информационные технологии, технику, прикладные программные средства при решении профессиональных задач; использовать языки программирования, строить логически правильные и эффективные программы; распознавать в конкретных прикладных задачах те или иные алгоритмы решения. Владеть: навыками использования современных информационных технологий в проектно-конструкторской деятельности; методами и приемами решения прикладных задач; методикой программирования этих задач.	лекции, самостоятельная работа, практические занятия	УО, К/Р, Т; Э	Базовый уровень: воспроизводство полученных знаний в ходе текущего контроля Повышенный уровень: владеет навыками использования современных информационных технологий в проектно-конструкторской деятельности

В процессе обучения используются следующие оценочные формы самостоятельной работы студентов, оценочные средства текущего контроля успеваемости и промежуточных аттестаций:

- индивидуальный опрос;
- выполнение индивидуальных практических заданий;
- выполнение контрольных работ на практических занятиях;
- тестирование по дисциплине;
- зачёт по дисциплине;
- экзамен по дисциплине.

Контрольные вопросы и задания для проведения промежуточной аттестации формирование компетенций ОПК-14

	Вопросы зачёта
1	Основные структуры алгоритма или программы.
2	Основные конструкции в программировании.
3	Язык Паскаль и среда разработки.
4	Основные операции над вещественными и символьными типами данных.
5	Множества и операции над множествами.
6	Составить программу нахождения средне-арифметического положительных элементов матрицы.
7	Массивы: одномерные и двумерные.
8	Классические задачи для одномерных массивов.
9	Классические задачи для двумерных массивов.
10	Графические задачи на Паскале.

формирование компетенций ОПК-14

	Вопросы экзамена
1	Что такое записи
2	Основные методы работы с записями.
3	Термины в объектно-ориентированном программировании.
4	Объекты и их свойства.
5	Объекты и их методы.
6	Создание класса объекта.
7	Введение в язык Си.
8	Типы данных в языке Си и операции.
9	Типовые задачи на языке Си.
10	Графика в Си.
11	Работа со строками в языке Си.
12	Составить программу на языке Си нахождения средне-арифметического положительных элементов матрицы.

Текущий контроль (контрольные работы) формирование компетенций ОПК-14

Тематика заданий и задач для текущего контроля	
«Массивы»	
1	Даны две матрицы 5x5. Данные матриц введены случайным образом. Построить третью матрицу: диагональ матрицы – нулевые элементы. Все, что выше диагонали – сумма соответствующих элементов двух матриц, а все, что ниже диагонали – разность соответствующих элементов.
2	Дана последовательность из 20 чисел. Числа заданы случайным образом в диапазоне (-10, 10). Найти количество четных чисел, количество нечетных, количество положительных и отрицательных чисел.
«Объектно-ориентированное программирование»	

1	Создать класс объекта «Покупатель». Описать его свойства и методы.
2	Описать класс для работы с одномерным массивом целых чисел (вектором). Класс должен реализовывать возможность: выполнение операций умножения и деления всех элементов массива на скаляр.
«Работа с массивами в Си»	
1	Дана последовательность из 20 элементов. Массив заполняется случайным образом. Найти первые 5 наибольших элементов. Вывести эти 5 наибольших элементов, расположив их в порядке убывания.
2	Дана матрица размером 10x10. Данные введены случайным образом в диапазоне (-10, 10). Посчитать количество отрицательных элементов, кратных 3, и количество положительных элементов, больших 6.
«Операции со строками. Сортировка»	
1	Составить программу на языке Си определения, является ли введенное слово палиндромом.
2	Составить на языке Си программу сортировки массива пузырьковым методом.

Критерии оценки контрольной работы

Оценка	Критерий оценки
Отлично	полное, правильное выполнение заданий с отдельными недочётами; выполнение от 90% и более.
Хорошо	правильное выполнение заданий с незначительным количеством ошибок; выполнение более 75% менее 95 %.
Удовлетворительно	выполнение основной части заданий с ошибками; выполнение более 50% менее 75 %.
Неудовлетворительно	частичное выполнение заданий (менее половины); допущение значительного количества ошибок; выполнение менее 50%.

Контрольные вопросы устный опрос

формирование компетенций ОПК-14

- 1 Алгоритм, его свойства, способы представления.
2. Программы на языках высокого уровня. Постановка задачи и спецификация программы. Основные этапы решения задач на ЭВМ.
3. Критерии качества программ; диалоговые программы; жизненный цикл программ.
4. Определение типа данных. Статический и динамический контроль типов и классификация языков в соответствии с ним.
5. Слабая, сильная и строгая типизации в статически типизируемых языках.
- 6 Конструкции структурного программирования и теорема о структурировании.
- 7 Абстракция управления.
- 8 Объект, как активный процесс. Основные понятия: инкапсуляция, наследование и полиморфизм. Наследование и переопределение.
- 9 Раннее и позднее связывание. Виртуальные методы. Конструкторы и деструкторы. Архитектура, управляемая событиями.
- 10 Лексическая структура языка. Структура паскаль-программ, разделы и их назначение.
- 11 Понятия типа, иерархия типа. Целые типы, литерный тип, вещественные типы. Генератор случайных чисел.
- 12 Булевский тип. Логические операции, логические функции.
- 13 Выражения, правила их записи и выполнения, классификация выражения по типу результата.
- 14 Ввод информации, вывод информации на экран. Форматный вывод.
- 15 Программирование ветвлений. Понятие цикла, циклы с заданным числом повторений и итерационные. Специальный выход из цикла.
- 16 Вложенные циклы, их организация с помощью всех трех операторов. Правила использова-

ния.

17 Перечислимый тип. Правила использования, действия с данными указанного типа

18 Ограниченный тип, области его применения. Функции low, high. Приведение типа переменной.

19 Одномерные, двумерные массивы, ввод/вывод, использование индексов нечислового типа. Ошибки в работе с регулярными типами, режимы компиляции. \$R+ и \$R-.

20 Строковый тип. Процедуры и функции для обработки строк (пример программы обработки текста с использованием строк).

21 Множества, действия с множествами.

22 Понятие структуры данных запись. Иерархические записи, массивы записи, оператор присоединения. Записи с вариантами, действия с ними, размещение в памяти.

23 Процедуры. Описание и вызов. Формальные и фактические параметры, их взаимодействие.

24 Параметры-переменные, параметры-значения, параметры-константы. Параметры сложных типов. Бестиповые параметры. Открытые параметры и строки.

25 Функции. Описание, вызов.

26 Область действия имен. Формальные и фактические параметры. Локальные и глобальные идентификаторы.

27 Процедурный и функциональный типы. Правила использования.

28 Рекурсивные алгоритмы. Порядок отведения памяти под локальные параметры. Пример программы.

29 Синтаксические анализаторы. Элементы металингвистических формул. Пример программы.

30 Абстракция модульности. Модульное программирование. Принцип скрытия информации. Аксиома модульности.

31 Принцип сборочного программирования. Определение модуля. Модули в языках программирования.

32 Процедурная абстракция. Основные характеристики программного модуля. Гипотеза о глобальных данных.

33 Размер модуля. Сцепление модулей. Связность модуля. Методы разработки структуры программы.

34 Структура модуля. Использование модулей. Пример с интерфейсной и реализующей частью.

35 Правила использования модулей. Пример модуля с разделом инициализации.

36 Компиляция модулей. 4. Процедуры и функции модуля CRT.TPU, STRINGS.TPU, DOS.TPU.

37 Файлы. Файловые типы. Понятие прямого и последовательного методов доступа.

38 Файловая система Турбо Паскаля: понятие логического и физического файлов.

39 Текстовые файлы.

40 Типизированные файлы. Процедуры и функции для организации прямого метода доступа.

41 Алгоритмы формирования, просмотра и корректировки текстового и типизированного файлов.

42 Бестиповые файлы.

43 Ссылочные типы: стандартный и пользовательский. Действия с указателями.

44 Динамические структуры данных. Схема распределения памяти при выполнении программ.

45 Действия с динамическими переменными: создание и уничтожение. Динамические массивы и массивы указателей. Функции для определения свободной динамической памяти.

46 Линейные списки. Формирование, просмотр, удаление и включение элемента. Примеры программ.

47 Стек. Формирование, добавление, исключение элемента из стека. Примеры программ.

48 Очереди. Включение элемента в очередь и удаление.

49 Двухнаправленные списки. Создание списка, удаление и вставка элемента.

50 Методы разработки структуры программы. Иерархические схемы. Правила детализации. Сегментирование.

51 Структурное и функциональное тестирование.

52 Тестирование модулей и интегральное тестирование. Формирование тестовых данных: классов, этапов тестирования, анализа результатов.

53 Тестовые мониторы. Виды отладки. Языки и системы тестирования. Отладчики и отладочная информация.

Критерии оценки текущего контроля

Оценка «отлично» выставляется студенту, если студент ориентируется в теоретическом материале; имеет представление об основных подходах к излагаемому материалу; знает определения основных теоретических понятий излагаемой темы, умеет применять теоретические сведения для анализа практического материала, в основном демонстрирует готовность применять теоретические знания в практической деятельности и освоение большинства показателей формируемых компетенций.

Оценка «хорошо» выставляется студенту, если студент ориентируется в теоретическом материале; имеет представление об основных подходах к излагаемому материалу, но затрудняется в ответах на некоторые вопросы; знает определения основных теоретических понятий излагаемой темы, но не в полной мере отражает суть рассматриваемой проблемы, в основном умеет применять теоретические сведения для анализа практического материала, в основном демонстрирует готовность применять теоретические знания в практической деятельности и освоение большинства показателей формируемых компетенций.

Оценка «удовлетворительно» выставляется студенту, если показаны недостаточные знания теоретического материала, основных понятий излагаемой темы, не всегда с правильным и необходимым применением специальных терминов, понятий и категорий; анализ практического материала был нечёткий.

Оценка «неудовлетворительно» выставляется в случаях, когда не выполнены условия, позволяющие выставить оценку «удовлетворительно».

Тест по дисциплине формирование компетенций ОПК-14

1.1 ЭТАПЫ РЕШЕНИЯ ЗАДАЧ НА ЭВМ

ТЗ_1 Последовательность основных этапов проектирования задач на ЭВМ

- 1 Постановка задачи
- 2 Построение математической модели
- 3 Разработка алгоритма
- 4 Программирование
- 5 Тестирование и отладка

ТЗ_2 Этап проектирования задачи на ЭВМ, на котором анализируется условие задачи, определяются исходные данные и результаты, устанавливается зависимость между величинами, рассматриваемыми в задаче, называется ...

- 1 постановка задачи ·
- 2 построение математической модели ·
- 3 разработка алгоритма ·
- 4 программирование ·
- 5 тестирование и отладка

ТЗ_3 Этап проектирования задачи на ЭВМ, на котором происходит запуск тестовой программы с использованием контрольных примеров, называется ...

- 1 постановка задачи ·
- 2 построение математической модели ·
- 3 разработка алгоритма ·
- 4 программирование ·
- 5 тестирование и отладка

ТЗ_4 Этап проектирования задачи на ЭВМ, на котором происходит формулировка условия задачи, называется ...

- 1 постановка задачи ·
- 2 построение математической модели ·
- 3 разработка алгоритма ·
- 4 программирование ·

5 тестирование и отладка

ТЗ_5 Этап проектирования задачи на ЭВМ, на котором программа записывается на компьютерном языке и вводится в ЭВМ, называется

1 постановка задачи .

2 построение математической модели .

3 разработка алгоритма .

4 программирование .

5 тестирование и отладка

ТЗ_6 Этап проектирования задачи на ЭВМ, на котором происходит проектирование и выбор существующего или разработка нового метода решения, называется

1 постановка задачи .

2 построение математической модели .

3 разработка алгоритма .

4 программирование .

5 тестирование и отладка

ТЗ_7 Этап проектирования задачи на ЭВМ, на котором создается математическая модель решения задачи, называется этапом

1 формализации .

2 алгоритмизации .

3 программирования .

4 тестирования .

5 отладки .

6 решения

ТЗ_8 Выбор метода решения задачи для ЭВМ происходит перед

1 разработкой алгоритма

2 построением математической модели .

3 анализом результатов .

4 тестированием и отладкой .

5 программированием

ТЗ_9 Этап, являющийся заключительным при решении задач на ЭВМ, –

1 построение математической модели .

2 анализ результатов .

3 тестирование и отладка .

4 программирование .

5 разработка алгоритма

ТЗ_10 На этапе тестирования и отладки при проектировании задачи на ЭВМ происходит

1 получение результата .

2 обнаружение ошибок .

3 запись алгоритма на языке программирования .

4 формализация задачи .

5 составление алгоритма решения задачи

ТЗ_11 Ошибки, при решении задачи на ЭВМ, которые не обнаруживаются транслятором, –
логические .

синтаксические .

динамические .

логические и синтаксические

1.2 ОСНОВНЫЕ ПОНЯТИЯ АЛГОРИТМИЗАЦИИ

ТЗ_12 Алгоритмом можно назвать .

1 рецепт приготовления пирога .

2 расписание занятий .

3 список покупок в магазине .

4 технический паспорт компьютера .

5 правила перевода из одной системы счисления в другую .

6 правила техники безопасности .

7 инструкцию по настройке телевизора

ТЗ_13 Алгоритмом является следующее описание

·Направо-налево равняйся! На первый-второй
рассчитайся! ·

1 Пойди туда, не знаю куда. Принеси то, не знаю что. ·

2 Возьми, что нужно. ·

3 Сделай как следует. Получишь то, что желаешь. ·

4 Сделай шаг вперед. Сделай шаг назад. Начни сначала. ·

5 0010101101 101001.

ТЗ_14 Заранее заданное понятное и точное предписание исполнителю совершить определённую последовательность действий для получения решения задачи за конечное число шагов называется ... ·

алгоритм

ТЗ_15 Алгоритм – это ... ·

1 некоторые истинные высказывания, которые должны быть направлены на достижение поставленной цели ·

2 отражение предметного мира с помощью знаков и сигналов, предназначенное для конкретного исполнителя ·

3 понятное и точное предписание исполнителю совершить определённую последовательность действий ·

4 представление кода программы на языке программирования ·

5 система инструкций для исполнителя

ТЗ_16 Исполнитель алгоритма – это ... ·

1 человек или компьютер, умеющий выполнять определённый набор действий ·

2 понятное и точное предписание необходимых действий ·

3 определённые условия выполнения действий ·

4 элемент, связывающий этапы выполнения алгоритма

ТЗ_17 Алгоритм, записанный на «понятном» компьютеру языке программирования, называется

1 исполнителем алгоритмов ·

2 программой ·

3 блок-схема ·

4 текстовкой ·

5 протоколом алгоритма

ТЗ_18 Верные утверждения: ... ·

1 алгоритм – это совокупность всех команд, которые могут быть выполнены исполнителем

2 исполнителем алгоритма может быть только компьютер ·

3 алгоритм может быть записан как в виде блок-схем, так и на языке программирования ·

4 исполнителем алгоритма, представленного в виде блок-схемы, является компьютер ·

5 исполнителем алгоритма, который записан на языке программирования, является человек

6 программа – это алгоритм, записанный на определённом языке программирования

ТЗ_19 Графическое задание алгоритма (блок-схемы) – это ... ·

1 способ представления алгоритма с помощью геометрических фигур ·

2 представление алгоритма в форме таблиц и расчётных формул ·

3 система обозначения правил для единообразной и точной записи алгоритмов их исполнения ·

4 схематичное изображение в произвольной форме

5 формализованная задача.

Критерии оценки:

отлично – от 90% до 100% правильных ответов;

хорошо – от 75% до 90% правильных ответов;

удовлетворительно – от 55% до 75% правильных ответов;

неудовлетворительно – менее 55% правильных ответов.

Форма экзаменационного билета

МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ АВТОНОМНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ОБРАЗОВАНИЯ
«МОСКОВСКИЙ ПОЛИТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ»
(МОСКОВСКИЙ ПОЛИТЕХ)

Электростальский институт (филиал)
Московского политехнического университета

Факультет **Управления**

Кафедра – Прикладной математики и информатики

Направление подготовки **27.03.04 Управление в технических системах**

Дисциплина – **Программирование и основы алгоритмизации**

Курс , семестр , группы _____, форма обучения: очная, очно-очно-заочная, очно-заочная

Экзаменационный билет № 1

- 1 Основные методы работы с записями.
- 2 Типы данных в языке Си и операции.
- 3 Составить на языке Си программу сортировки массива пузырьковым методом.

Приложение В к рабочей программе
7.3 Оценочные средства по дисциплине

«Программирование и основы алгоритмизации»

№ ОС	Наименование оценочного средства	Краткая характеристика оценочного средства	Представление оценочного средства в ФОС
1	Контрольная работа (К/Р)	Средство проверки умений применять полученные знания для решения задач определенного типа по теме или разделу.	Комплект контрольных заданий по вариантам
2	Устный опрос собеседование (УО)	Средство контроля, организованное как специальная беседа педагогического работника с обучающимся на темы, связанные с изучаемой дисциплиной, и рассчитанное на выяснение объема знаний, обучающегося по определенному разделу, теме, проблеме и т.п.	Вопросы по темам/разделам дисциплины
3	Тест (Т)	Система стандартизированных заданий, позволяющая автоматизировать процедуру измерения уровня знаний и умений обучающегося.	Фонд тестовых заданий
4	Зачёт (З)	Итоговая форма оценки знаний. В высших учебных заведениях проводится во время сессии.	Вопросы к зачёту
5	Экзамен (Э)	Итоговая форма оценки знаний. В высших учебных заведениях проводится во время сессии.	Вопросы к экзамену