

МИНОБРНАУКИ РОССИИ

Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
**«Тверской государственный технический университет»**  
(ТвГТУ)

УТВЕРЖДАЮ

Проректор по учебной работе  
\_\_\_\_\_ Э.Ю. Майкова

« \_\_\_\_ » \_\_\_\_\_ 2019 г.

**РАБОЧАЯ ПРОГРАММА**

дисциплины обязательной части Блока 1 «Дисциплины (модули)»  
**«Алгоритмические языки и программирование»**

Направление подготовки бакалавров – 09.03.01 Информатика и вычислительная техника

Направленность (профиль) – Вычислительные машины, комплексы, системы и сети

Тип задач профессиональной деятельности – производственно-технологическая

Форма обучения – очная и заочная

Факультет информационных технологий

Кафедра электронных вычислительных машин

Тверь 2019

Рабочая программа дисциплины соответствует ОХОП подготовки бакалавров в части требований к результатам обучения по дисциплине и учебному плану.

Разработчик программы: ст.преподаватель

А.Н. Неведомский

Программа рассмотрена и одобрена на заседании кафедры ЭВМ 29.03.2019 г. протокол № 5.

Заведующий кафедрой ЭВМ

А.Р. Хабаров

Согласовано:

Начальник УМО УМУ

Д.А. Барчуков

Начальник отдела  
комплектования  
зональной научной  
библиотеки

О.Ф. Жмыхова

## 1. Цель и задачи дисциплины

**Основной целью** изучения дисциплины «Алгоритмические языки и программирование» является расширение мировоззрения студентов и освоение общих принципов и средств, необходимых для разработки программного обеспечения электронных вычислительных машин.

**Задачи курса:** изучение основных принципов и правил алгоритмизации; изучение основных принципов программирования на языке высокого уровня С# и получение практических навыков разработки программного обеспечения на этом языке в среде Microsoft Visual Studio; изучение сложных динамических структур данных и получение практических навыков их использования при разработке программ.

## 2. Место дисциплины в структуре ОП

Дисциплина относится к обязательной части Блока 1 ОП ВО. Для изучения курса требуются знания дисциплины «Информатика», полученных при обучении в средней школе.

Знания, полученные при изучении данной дисциплины, используются студентами при изучении дисциплин «Технологии программирования», «Системное программное обеспечение», «Технологии удалённого взаимодействия» и других.

## 3. Планируемые результаты обучения по дисциплине

### 3.1. Планируемые результаты обучения по дисциплине

#### **Компетенции, закреплённые за дисциплиной в ОХОП:**

**УК-1.** *Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач.*

#### **Индикатор компетенции, закреплённый за дисциплиной в ОХОП:**

ИУК-1.2. Выполняет поиск необходимой информации, её критический анализ и обобщает результаты анализа для решения поставленной задачи.

#### **Показатели оценивания индикатора достижения компетенции:**

ИУК-1.2.:

#### **Знать:**

З1: Теоретический фундамент, необходимый для решения задач в области проектирования прикладного программного обеспечения.

#### **Уметь:**

У1: Применять основные современные инструментальные средства и программное обеспечение в области разработки программного обеспечения.

**ОПК-8.** *Способен разрабатывать алгоритмы и программы, пригодные для практического применения.*

**Индикаторы компетенции, закреплённый за дисциплиной в ОХОП:**

ИОПК-8.1. Анализирует особенности предметной области, формулирует задачи.

ИОПК-8.2. Выбирает и обосновывает метод решения задачи и разрабатывает алгоритм.

ИОПК-8.3. Разрабатывает программу в одной из сред программирования.

**Показатели оценивания индикатора достижения компетенции:**

ИОПК-8.1:

**Знать:**

З1: Методы использования базовых положений математики в алгоритмах решения задач программирования.

З2: Понятие о системном и прикладном программном обеспечении и методах их разработки.

**Уметь:**

У1: Ставить задачу при решении профессиональных задач, определять наборы входных и выходных данных.

ИОПК-8.2:

**Знать:**

З1: Требования, предъявляемые к алгоритмам. Технологию разработки алгоритмов и программ на алгоритмическом языке.

**Уметь:**

У1: Разрабатывать алгоритмы и программы на алгоритмическом языке.

ИОПК-8.3:

**Знать:**

З1. Современные технологии разработки прикладных программ и программных комплексов.

З.2. Основные алгоритмические конструкции и правила их использования при разработке алгоритма решения задачи.

З.3. Основные синтаксические конструкции языка программирования высокого уровня C# и правила записи программы на основе разработанного алгоритма.

**Уметь:**

У.1. Формализовать поставленную задачу и довести ее до рабочего алгоритма.

У.2. Создавать прикладные программы, использующие различные типы интерфейса пользователя (консольное приложение, приложение с графическим интерфейсом на Windows Forms).

### 3.2. Технологии, обеспечивающие формирование компетенций

Проведение лекционных занятий, выполнение лабораторных работ, практических занятий и курсовой работы, самостоятельная работа.

#### 4. Трудоемкость дисциплины и виды учебной работы

Таблица 1а. Распределение трудоемкости дисциплины по видам учебной работы

##### ОЧНАЯ ФОРМА ОБУЧЕНИЯ

Вид учебной работы	Зачетные единицы	Академические часы
<b>Общая трудоемкость дисциплины</b>	<b>8</b>	<b>288</b>
<b>Аудиторные занятия (всего)</b>		120
В том числе:		
Лекции		60
Практические занятия (ПЗ)		15
Лабораторные работы (ЛР)		45
<b>Самостоятельная работа обучающихся (всего)</b>		96+72 (экз.)
В том числе:		
Курсовая работа		70
Курсовой проект		не предусмотрен
Расчетно-графические работы		не предусмотрены
Реферат		не предусмотрен
Другие виды самостоятельной работы: - подготовка к защите лабораторных работ - подготовка к практическим занятиям		13 13
Текущий контроль успеваемости и промежуточная аттестация (экзамен, экзамен)	2	72 (экз.)
<b>Практическая подготовка при реализации дисциплины (всего)</b>		0

Таблица 1б. Распределение трудоемкости дисциплины по видам учебной работы

##### ЗАОЧНАЯ ФОРМА ОБУЧЕНИЯ

Вид учебной работы	Зачетные единицы	Академические часы
<b>Общая трудоемкость дисциплины</b>	<b>8</b>	<b>288</b>
<b>Аудиторные занятия (всего)</b>		20
В том числе:		
Лекции		10
Практические занятия (ПЗ)		2
Лабораторные работы (ЛР)		8
<b>Самостоятельная работа обучающихся (всего)</b>		268

В том числе:		
Курсовая работа		70
Курсовой проект		не предусмотрен
Расчетно-графические работы		не предусмотрены
Реферат		не предусмотрен
Другие виды самостоятельной работы: - изучение теоретической части дисциплины; - подготовка к защите лабораторных работ - подготовка практических работ		150 24 24
Текущий контроль успеваемости и промежуточная аттестация (экзамен, экзамен)		18
<b>Практическая подготовка при реализации дисциплины (всего)</b>		0

## 5. Структура и содержание дисциплины

### 5.1. Структура дисциплины

#### ОЧНАЯ ФОРМА ОБУЧЕНИЯ

Таблица 2а. Модули (разделы) дисциплины, трудоемкость в часах и виды учебной работы

№	Наименование модуля	Трудо-сть часы	Лекции	Практич. занятия	Лабор. практи- кум	Сам. работа
1	Язык программирования С# как язык высокого уровня	32	6	3	4	10+9
2	Алгоритмизация и обработка числовых массивов	41	10	4	8	10+9
3	Обработка символьных типов данных. Создание и использование собственных функций	35	6	4	6	10+9
4	Работа с файлами данных	34	7	4	6	8+9
5	Основы создания и использования динамических структур данных	30	7	–	4	10+9
6	Создание и использование динамических структур на основе списков	44	8	–	7	20+9
7	Создание и использование двоичных деревьев и графов	37	8	–	6	14+9
8	Основы разработки графического интерфейса	35	8	–	4	14+9
	Всего на дисциплину	288	60	15	45	96+72 (экз.)

#### ЗАОЧНАЯ ФОРМА ОБУЧЕНИЯ

Таблица 2б. Модули (разделы) дисциплины, трудоемкость в часах и виды учебной работы

№	Наименование модуля	Трудо-сть часы	Лекции	Практич. занятия	Лабор. практи- кум	Сам. работа
1	Язык программирования С# как язык высокого уровня	31,5	0,5	-	1	30
2	Алгоритмизация и обработка числовых массивов	38	2	-	1	35
3	Обработка символьных типов данных. Создание и использование собственных функций	36	1	1	1	33
4	Работа с файлами данных	36	1	1	1	33
5	Основы создания и использования динамических структур данных	35	1	-	1	33
6	Создание и использование динамических структур на основе списков	37,5	1,5	-	1	35
7	Создание и использование двоичных деревьев и графов	38	2	-	1	35
8	Основы разработки графического интерфейса	36	1	-	1	34
	Всего на дисциплину	288	10	2	8	268

## 5.2. Содержание дисциплины

### Модуль 1. Язык программирования С# как язык высокого уровня

Введение. Языки программирования низкого, среднего и высокого уровней. Основные этапы написания программы. Пример простой программы на языке С#. Классы ввода и вывода. Пространства имен и функция Main().

Типы данных и операции в языке С#. Преобразование типов – явные и неявные. Управляющие структуры в языке С# – ветвление и выбор. Организация циклов. Использование операторов break и continue. Оператор перехода по метке и его использование. Структура библиотеки классов .NET. Структура проекта и назначение его файлов. Различные типы проектов, создаваемые на языке С# на платформе .NET.

### Модуль 2. Алгоритмизация и обработка числовых массивов

Этапы разработки программы. Алгоритмизация и способы разработки алгоритмов. Блок-схема и псевдокод. Основные алгоритмические конструкции и их запись в алгоритме. Метод последовательной детализации и его использование.

Одномерные массивы и работа с ними. Основные задачи, решаемые с помощью массивов и алгоритмы их решения. Обработка элементов массива – полная и

по заданному условию. Поиск данных в массиве. Удаление и вставка элементов массива. Упорядочивание массива.

Матрицы и работа с ними. Основные задачи, решаемые с помощью матриц и алгоритмы их решения. Обработка матриц по строкам и по столбцам. Удаление и вставка строк и столбцов матриц. Квадратные матрицы и работа с ними.

Невыровненные (ступенчатые) двумерные массивы и их использование. Многомерные массивы и их использование.

### **Модуль 3. Обработка символьных типов данных. Создание и использование собственных функций**

Строки символов и работа с ними. Основные задачи, решаемые со строками символов. Разбиение строки на слова и работа с отдельными символами. Стандартные функции C# для работы со строками и их использование.

Технология создания больших программ. Разработка собственных функций. Параметры и возвращаемые значения функций. Передача параметров функции по ссылке и значению. Функциональное программирование. Рекурсивные алгоритмы и функции и их использование. Области действия имен переменных.

Создание собственных типов данных. Структуры и перечисления, их использование. Создание и использование массивов структур. Обращение к элементам структуры через структурную переменную.

### **Модуль 4. Работа с файлами данных**

Обработка файлов данных. Понятие файла данных на внешнем носителе. Текстовые и двоичные файлы. Основные операции с файлами. Стандартные классы и функции для работы с файлами.

Стандартные приемы обработки текстового файла по строкам и по символам. Использование структур для обработки текстовых файлов. Стандартные приемы обработки двоичного файла по записям. Использование структур для обработки двоичных файлов.

Использование двоичных файлов для хранения изображений. Структура файла изображения на примере BMP. Выполнение операций над изображениями в двоичном файле.

Связь программы с операционной системой. Использование функций файловой системы ОС. Программное выполнение системных команд. Работа с процессами операционной системы.

### **Модуль 5. Основы создания и использования динамических структур данных**

Указатели и работа с ними. Переменные, константы и выражения типа указатель. Операции получения адреса и взятия значения по указателю. Операции увеличения/уменьшения указателей и их специфика. Использование указателей для работы со стековыми массивами и матрицами.



Динамическое выделение памяти и сборка мусора. Создание динамического одномерного и двумерного стекового массива и используемые стандартные функции. Использование динамических массивов и матриц.

Создание динамической матрицы через указатель на указатель. Использование указателей на различные типы данных. Использование указателей для совместимости с функциями API Windows.

## **Модуль 6. Создание и использование динамических структур на основе списков**

Связные списки и операции с ними. Использование стандартных классов ArrayList и List<type> для организации списков.

Односвязные, двусвязные и кольцевые списки. Создание и использование собственного класса для реализации различных типов списков.

Стеки, деки и очереди и их реализация на основе списков.

Алгоритмы с использованием стеков и очередей и их реализация.

Использование стеков и очередей в операционных системах. Передача параметров функции и возврат значений с использованием стека. Организация очередей для использования центрального процессора и доступа к дисковым устройствам в мультипрограммных ОС.

## **Модуль 7. Создание и использование двоичных деревьев и графов**

Деревья и операции с ними. Двоичные деревья: упорядоченные, неупорядоченные, сбалансированные. Организация быстрого поиска информации в дереве. Сильноветвящиеся деревья. Основные задачи, решаемые с помощью деревьев, и алгоритмы их решения.

Графы и работа с ними. Логическая структура и машинное представление ориентированных и неориентированных графов. Основные задачи, решаемые с помощью графов, и алгоритмы их решения.

Методы ускорения доступа к данным. Создание хэш-таблиц и их использование. Методы разрешения коллизий. Оценка качества хэш-функций. Инвертированные индексы и битовые карты.

## **Модуль 8. Основы разработки графического интерфейса**

Основы построения графических изображений и использования графического интерфейса Windows. Понятие события и создание обработчиков событий мыши и клавиатуры.

Основные интерфейсные объекты Windows Forms (надписи, поля ввода, кнопки, списки, панели) и их использование.

Класс Graphics и его использование для создания графических изображений. Рисование элементарных графических изображений (линий, прямоугольников, эллипсов, дуг) в окне программы. Событие OnPaint() и его обработка.

Использование дополнительных объектов графического интерфейса для рисования. Создание классов, имеющих собственные функции отрисовки, для представления данных пользователя.

### 5.3 Лабораторные работы

#### ОЧНАЯ ФОРМА ОБУЧЕНИЯ

Таблица 3а. Тематика, форма лабораторных работ (ЛР) и их трудоемкость

Порядковый номер модуля. Цели лабораторных работ	Наименование лабораторных работ	Трудоемкость в часах
Модуль 1. Цель: формирование умений и навыков работы с интегрированной средой MS Visual Studio для разработки и выполнения простых программ на C#. Изучение основных типов данных, операций с ними и управляющих конструкций «Ветвление»	Ознакомление со средой программирования MS Visual Studio	2
	Вычисление арифметического выражения	2
Модуль 2. Цель: формирование умений и навыков работы с одномерными и двумерными массивами данных. Изучение и практическое использование конструкций «Повторение» различного типа, основных алгоритмов сортировки данных	Обработка одномерных массивов чисел	3
	Обработка числовых матриц	5
Модуль 3. Цель: формирование умений и навыков работы с символьными и строковыми данными, а также разработки собственных функций, в том числе рекурсивных	Работа с символьными строками Разработка собственных функций	3
	Рекурсивные функции	3
Модуль 4. Цель: формирование умений и навыков работы с файловой системой, а также обработки данных из файлов различных типов	Обработка тестовых файлов	3
	Обработка двоичных файлов	3
Модуль 5. Цель: формирование умений и навыков работы со стековыми массивами и указателями в C#, а также использования	Использование стековых массивов и указателей	4
Модуль 6. Цель: формирование умений и навыков работы со списковыми структурами данных, а также формирование собственных структур данных (очереди и стеков) на основе списков	Обработка списков	3
	Создание структур данных на основе списков	4
Модуль 7. Цель: формирование умений и навыков работы с двоичными деревьями и графами	Обработка двоичных деревьев	3
	Обработка графов	3

Модуль 8. Цель: формирование умений и навыков создания приложений с графическим интерфейсом и отрисовки собственных сложных объектов	Создание программ с графическим интерфейсом	4
---	---	---

### ЗАОЧНАЯ ФОРМА ОБУЧЕНИЯ

Таблица 3б. Тематика, форма лабораторных работ (ЛР) и их трудоемкость

Порядковый номер модуля. Цели лабораторных работ	Наименование лабораторных работ	Трудоемкость в часах
Модуль 1. Цель: формирование умений и навыков работы с интегрированной средой MS Visual Studio для разработки и выполнения простых программ на C#. Изучение основных типов данных, операций с ними и управляющих конструкций «Ветвление»	Ознакомление со средой программирования MS Visual Studio	0,5
	Вычисление арифметического выражения	0,5
Модуль 2. Цель: формирование умений и навыков работы с одномерными и двумерными массивами данных. Изучение и практическое использование конструкций «Повторение» различного типа, основных алгоритмов сортировки данных	Обработка одномерных массивов чисел	0,5
	Обработка числовых матриц	0,5
Модуль 3. Цель: формирование умений и навыков работы с символьными и строковыми данными, а также разработки собственных функций, в том числе рекурсивных	Работа с символьными строками Разработка собственных функций	0,5
	Рекурсивные функции	0,5
Модуль 4. Цель: формирование умений и навыков работы с файловой системой, а также обработки данных из файлов различных типов	Обработка тестовых файлов	0,5
	Обработка двоичных файлов	0,5
Модуль 5. Цель: формирование умений и навыков работы со стековыми массивами и указателями в C#, а также использования	Использование стековых массивов и указателей	1
Модуль 6. Цель: формирование умений и навыков работы со списковыми структурами данных, а также формирование собственных структур данных (очереди и стеков) на основе списков	Обработка списков	0,5
	Создание структур данных на основе списков	0,5
Модуль 7. Цель: формирование умений и навыков работы с двоичными деревьями и графами	Обработка двоичных деревьев	0,5
	Обработка графов	0,5

Модуль 8. Цель: формирование умений и навыков создания приложений с графическим интерфейсом и отрисовки собственных сложных объектов	Создание программ с графическим интерфейсом	1
---	---	---

#### 5.4. Практические занятия

##### ОЧНАЯ ФОРМА ОБУЧЕНИЯ

Таблица 4а. Тематика, форма практических занятий (ПЗ) и их трудоемкость

Порядковый номер модуля. Цели практических занятий	Примерная тематика занятий и форма их проведения	Трудоемкость в часах
Модуль 1. Цель: Изучение основных типов данных, операций с ними и управляющих конструкций «Ветвление»	Основные типы данных и операции с ними	1,5
	Управляющие конструкции «Ветвление» и их использование	1,5
Модуль 2. Цель: Изучение и практическое использование конструкций «Повторение» различного типа для работы с одномерными и двумерными массивами, основных алгоритмов сортировки данных	Управляющие конструкции «Повторение» и их использование. Алгоритмы сортировки одномерных массивов	2
	Двумерные массивы и их обработка по строкам и столбцам	2
Модуль 3. Цель: изучение алгоритмов работы с символьными и строковыми данными, а также способов разработки собственных функций, в том числе рекурсивных	Символьные и строковые данные и их обработка	2
	Создание собственных функций и организация их взаимодействия	2
Модуль 4. Цель: изучение способов работы с файловой системой, а также обработки данных из файлов различных типов	Обработка текстовых файлов: используемые классы и функции	2
	Обработка двоичных файлов: используемые классы и функции	2

##### ЗАОЧНАЯ ФОРМА ОБУЧЕНИЯ

Таблица 4б. Тематика, форма практических занятий (ПЗ) и их трудоемкость

Порядковый номер модуля. Цели практических занятий	Примерная тематика занятий и форма их проведения	Трудоемкость в часах
Модуль 3. Цель: изучение алгоритмов работы с символьными и строковыми данными, а также способов разработки собственных функций, в том числе рекурсивных	Символьные и строковые данные и их обработка	0,5
	Создание собственных функций и организация их взаимодействия	0,5
Модуль 4. Цель: изучение способов работы с файло-	Обработка текстовых файлов: используемые классы и функции	0,5

вой системой, а также обработки данных из файлов различных типов	Обработка двоичных файлов: используемые классы и функции	0,5
--	--	-----

## 6. Самостоятельная работа обучающихся и текущий контроль успеваемости

### 6.1. Цели самостоятельной работы

Формирование способностей к самостоятельному познанию и обучению, поиску литературы, обобщению, оформлению и представлению полученных результатов, их критическому анализу, поиску новых и неординарных решений, аргументированному отстаиванию своих предложений, умений подготовки выступлений и ведения дискуссий.

### 6.2. Организация и содержание самостоятельной работы

Самостоятельная работа заключается в проработке отдельных тем курса по заданию преподавателя по рекомендованной им учебной и научной литературе, методическим рекомендациям кафедры, подготовке к экзамену, в выполнении курсовой работы.

В рамках дисциплины выполняется 14 лабораторных работ по очной форме обучения и 14 лабораторных работ по заочной форме, которые защищаются посредством устного опроса. Максимальная оценка за каждую выполненную лабораторную работу – 5 баллов, минимальная – 3 балла.

Выполнение всех лабораторных работ обязательно, в случае невыполнения лабораторной работы по уважительной причине студент имеет право выполнить работу с другой группой или во время защиты лабораторных работ.

После вводных лекций, в которых обозначается содержание дисциплины, ее проблематика и практическая значимость, студентам выдаются темы курсовой работы, определяется порядок подготовки доклада и презентации для ее защиты. Курсовая работа выполняется в соответствии с методическими указаниями по выполнению курсовой работы, разработанными на кафедре ЭВМ.

## 7. Учебно-методическое и информационное обеспечение дисциплины

### 7.1. Основная литература дисциплины

1. Паронджанов, В. Д. Алгоритмические языки и программирование: ДРАКОН : учебное пособие для вузов / В. Д. Паронджанов. — Москва : Издательство Юрайт, 2022. — 436 с. — (Высшее образование). — ISBN 978-5-534-13146-8. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/497311> . - (ID=139233-0).

2. Токманцев Т.Б. Алгоритмические языки и программирование : учебное пособие / Токманцев Т.Б.. — Екатеринбург : Уральский федеральный университет, ЭБС АСВ, 2013. — 104 с. — ISBN 978-5-7996-1023-4. — Текст : электронный // IPR

SMART : [сайт]. — URL: <https://www.iprbookshop.ru/68220.html>.

## 7.2. Дополнительная литература

1. Гагарина, Л.Г. Введение в теорию алгоритмических языков и компиляторов : учеб. пособие для вузов / Л.Г. Гагарина, Е.В. Кокорева. - М. : Форум, 2011. - 175 с. - (Высшее образование). - Текст : непосредственный. - ISBN 978-58199-0404-6 : 220 p. - (ID=85473-9).
2. Федоров, Д. Ю. Программирование на языке высокого уровня Python : учебное пособие для вузов / Д. Ю. Федоров. — 3-е изд., перераб. и доп. — Москва : Издательство Юрайт, 2022. — 210 с. — (Высшее образование). — ISBN 978-5-534-14638-7. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/492920> . - (ID=145296-0).
3. Гниденко, И.Г. Технологии и методы программирования: учебное пособие для вузов / И.Г. Гниденко, Ф.Ф. Павлов, Д.Ю. Федоров. - Москва : Юрайт, 2022. - Образовательная платформа Юрайт. - Текст : электронный. - ISBN 978-5-534-02816-4. - URL: <https://urait.ru/bcode/489920>. - (ID=145294-0).
4. Баженова, И.Ю. Языки программирования : учебник для вузов по напр. бакалавриата 010300 «Фундаментальная информатика и информационные технологии», 090900 «Информационная безопасность». / И.Ю. Баженова; под ред. В.А. Сухомлина. - Москва : Академия, 2012. - 358 с. - (Высшее профессиональное образование. Бакалавриат). - Текст : непосредственный. - ISBN 978-5-7695-6856-5 : 1127 p. 72 к. - (ID=93945-8).
5. Токарева, М.А. Введение в алгоритмизацию и программирование на языке C# : учебное пособие по направлению подготовки 09.03.02 «Информационные системы и технологии» : в 2 частях. Часть 1 / М.А. Токарева, М.И. Глотова, О.В. Приходько; Оренбургский государственный университет. - Оренбург: Оренбургский государственный университет, 2018. - ЭБС Лань. - Текст : электронный. - ISBN 978-5-7410-1998-6. - URL: <https://e.lanbook.com/book/159705>. - (ID=145881-0).
6. Павловская, Т.А. C/C++. Программирование на языке высокого уровня : учебник для бакалавров и магистров по напр. "Информатика и выч. техника" / Т.А. Павловская. - СПб. : Питер, 2011. - 460 с. - (Учебник для вузов). - Текст : непосредственный. - 299 p. 90 к. - (ID=87575-24).
7. Павловская, Т.А. C #. Программирование на языке высокого уровня: учебник для вузов для вузов по напр. подготовки дипломир. специалистов «Информатика и вычислительная техника» : в составе учебно-методического комплекса / Т.А. Павловская. - СПб. [и др.] : Питер, 2009. - 432 с. : ил. - (Учебник для вузов) (УМК-У). - Библиогр.: с. 425-426. - Текст : непосредственный. - ISBN 978-5-91180-174-8 : 209 p. - (ID=73911-7).
8. Мальков, А.А. Алгоритмизация и программирование на языках высокого уровня: учебное пособие / А.А. Мальков, Н.К. Жиганов, Г.П. Виноградов. - 1-е изд. - Тверь : ТвГТУ, 2006. - Сервер. - Текст : электронный. - ISBN 5-7995-0342-2 : 0-00. - URL: <http://elib.tstu.tver.ru/MegaPro/GetDoc/Megapro/61198>. - (ID=61198-1).

### 7.3. Методические материалы

1. Методические указания к курсовой работе по дисциплине «Алгоритмические языки и программирование». Направление подготовки бакалавров - 09.03.01 Информатика и вычислительная техника. Профиль - Вычислительные машины, комплексы, системы и сети : в составе учебно-методического комплекса / Каф. Электронно-вычислительные машины; сост. А.Н. Неведомский. - Тверь: ТвГТУ, 2017. - (УМК-М). - Сервер. - Текст: электронный. - (ID=124265-0).

2. Экзаменационные вопросы по дисциплине «Алгоритмические языки и программирование». Направление подготовки бакалавров - 09.03.01 Информатика и вычислительная техника. Профиль - Вычислительные машины, комплексы, системы и сети. Семестры 1, 2: в составе учебно-методического комплекса / Каф. Электронно-вычислительные машины; сост. А.Н. Неведомский. - Тверь: ТвГТУ, 2017. - (УМК-В). - Сервер. - Текст: электронный. - (ID=124263-0).

### 7.4. Программное обеспечение по дисциплине

1. Операционная система Microsoft Windows: лицензии № ICM-176609 и № ICM-176613 (Azure Dev Tools for Teaching).

2. Microsoft Office 2019 Russian Academic: OPEN No Level: лицензия № 41902814.

### 7.5. Специализированные базы данных, справочные системы, электронно-библиотечные системы, профессиональные порталы в Интернет

ЭБС и лицензионные ресурсы ТвГТУ размещены:

1. Ресурсы: <https://lib.tstu.tver.ru/header/obr-res>
2. ЭКТвГТУ: <https://elib.tstu.tver.ru/MegaPro/Web>
3. ЭБС «Лань»: <https://e.lanbook.com/>
4. ЭБС «Университетская библиотека онлайн»: <https://www.biblioclub.ru/>
5. ЭБС «IPRBooks»: <https://www.iprbookshop.ru/>
6. Электронная образовательная платформа «Юрайт» (ЭБС «Юрайт»): <https://urait.ru/>
7. Научная электронная библиотека eLIBRARY: <https://elibrary.ru/>
8. Информационная система «ТЕХНОРМАТИВ». Конфигурация «МАКСИМУМ» : сетевая версия (годовое обновление): [нормативно-технические, нормативно-правовые и руководящие документы (ГОСТы, РД, СНИПы и др.]. Диск 1,2,3,4. - М. :Технорматив, 2014. - (Документация для профессионалов). - CD. - Текст : электронный. - 119600 р. – (105501-1).
9. База данных учебно-методических комплексов: <https://lib.tstu.tver.ru/header/umk.html>.

УМК размещен: <https://elib.tstu.tver.ru/MegaPro/GetDoc/Megapro/111693>.

## **8. Материально-техническое обеспечение**

Кафедра электронных вычислительных машин имеет аудитории для проведения лекционных и лабораторных занятий по дисциплине; специализированный учебный класс для проведения компьютерных практикумов и самостоятельной работы, оснащенный современной компьютерной и офисной техникой, необходимым программным обеспечением, электронными учебными пособиями и законодательно-правовой поисковой системой, имеющий безлимитный выход в глобальную сеть.

## **9. Фонд оценочных средств для проведения промежуточной аттестации**

### **9. Оценочные средства для проведения промежуточной аттестации**

#### **9.1. Оценочные средства для проведения промежуточной аттестации в форме экзамена**

1. Шкала оценивания промежуточной аттестации в форме экзамена – «отлично», «хорошо», «удовлетворительно», «неудовлетворительно».

2. Критерии оценки и ее значения:

Для показателя «знать» (количественный критерий):

отсутствие знаний – 0 баллов,

наличие знаний – 2 балла.

Для показателя «уметь» (количественный критерий):

отсутствие умения – 0 баллов,

выполняет типовые задания с использованием стандартных алгоритмов – 1 балл,

выполняет усложненные задания на основе оригинальных алгоритмов решения или комбинации стандартных алгоритмов решения – 2 балла.

Критерии оценки за экзамен приводятся в экзаменационном билете.

3. Вид экзамена – письменный экзамен.

4. Форма экзаменационного билета.

Билет соответствует утвержденной Положением о рабочих программах дисциплин, соответствующих ФГОС ВО, форме. Типовой образец экзаменационного билета приведен в Приложении. Обучающемуся даётся право выбора заданий из числа, содержащихся в билете, принимая во внимание оценку, на которую он претендует.

С целью повышения ответственности обучающегося за результат экзамена устанавливаются следующие требования:

частично правильные ответы с дробными баллами не предусмотрены;

верное выполнение задания (решения задачи) не допускает любых погрешностей по существу задания.

5. База заданий, предназначенных для предъявления студентам на экзамене.



1. Понятие языка программирования. Уровни языков программирования. Языки программирования высокого уровня. Языки C, C++ и C#.
2. Платформено-независимые языки и .NET Framework. Структура простого консольного приложения на языке C#.
3. Переменные в C#. Имена переменных и их значения. Числовые литералы (константы). Символьные и строковые литералы (константы).
4. Типы данных в C#. Беззнаковые целые числа и операции над ними. Числа со знаком и операции над ними. Числа с плавающей точкой и для работы с денежными суммами.
5. Типы данных в C#. Символьный и логические типы и операции над ними. Инициализация переменных и оператор присваивания.
6. Ввод данных с консоли. Вывод данных на консоль. Использование классов Console и Convert.
7. Бинарные математические операции и их использование. Приоритет операций. Операция получения остатка от деления. Составные бинарные операции и их использование.
8. Унарные операции и их использование. Операции увеличения и уменьшения. Поразрядные логические операции и их использование. Операции сравнения (отношения) и логические операции.
9. Понятие алгоритма и исполнителя. Примеры исполнителей. Формы записи алгоритма.
10. Необходимые свойства алгоритма. Понятие о временной сложности алгоритма.
11. Реализация алгоритмических конструкций в C#: следование. Реализация алгоритмических конструкций в C#: ветвление. Вложенные условия.
12. Реализация алгоритмических конструкций в C#: выбор.
13. Реализация алгоритмических конструкций в C#: Цикл-Пока. Цикл-Делать-Пока.
14. Реализация алгоритмических конструкций в C#: Прерывание и возобновление цикла. Цикл с параметром.
15. Различные способы реализации цикла с параметром в C#. Цикл *foreach()* – область применения и специфика использования.
16. Одномерные массивы чисел – объявление, выделение памяти и инициализация.
17. Текстовые строки: описание, инициализация и базовые операции. Стандартные методы класса *String*.
18. Описание и использование собственных функций Передача параметров и возврат результата работы функции.
19. Передача параметров по ссылке и по значению. Выходные параметры функций и их использование.
20. Область действия имен переменных.
21. Файлы и потоки – работа со стандартными потоками ввода и вывода. Создание и использование потоков, связанных с текстовыми файлами
22. Создание и использование потоков, связанных с двоичными файлами. Перечисления, используемые для работы с двоичными файлами.

23. Обработка массивов: поиск последовательности элементов.
24. Обработка массивов: перестановка элементов в массиве.
25. Обработка массивов: сдвиг элементов массива.
26. Обработка массивов: удаление и вставка элементов.
27. Обработка массивов: упорядочение элементов (любой алгоритм).
28. Обработка массивов: создание одного массива из другого.
29. Обработка массивов: поиск одинаковых и различных элементов массива.
30. Многомерные массивы: объявление, выделение памяти и инициализация.
31. Обработка двумерных массивов по строкам.
32. Обработка двумерных массивов по столбцам.
33. Удаление строк или столбцов двумерного массива.
34. Вставка строк или столбцов в двумерный массив.
35. Квадратные матрицы и их обработка.
36. Дана целочисленная прямоугольная матрица. Поменять местами первую и последнюю строки матрицы, которые содержат одинаковые элементы.
37. Дана целочисленная прямоугольная матрица. Поменять местами первый и последний столбцы матрицы, которые не содержат одинаковых элементов.
38. Дана целочисленная прямоугольная матрица. Определить номер строки матрицы, которая содержит максимальное количество одинаковых элементов.
39. Дана матрица  $A[n,n]$ . Поменять местами столбцы, содержащие минимальный и максимальный элементы главной диагонали этой матрицы.
40. Дана матрица  $A[n,n]$ . Поменять местами строки, содержащие минимальные элементы главной и побочной диагоналей.
41. Дана матрица  $A[m,n]$ , выполнить циклический сдвиг вправо на одну позицию элементов каждой строки, находящихся между минимальным и максимальным элементом этой строки.
42. Дана матрица  $A[n,n]$ , упорядочить по возрастанию элементы главной диагонали.
43. Напишите программу, которая находит первое положительное число матрицы и меняет местами строку, в которой находится это число с последней строкой матрицы.
44. Дана матрица  $A[m,n]$ , выполнить перенос максимального элемента каждого столбца в первую строку этого же столбца, сдвинув элементы, находившиеся выше него на один вниз.
45. Дана матрица  $A[m,n]$ , выполнить циклический сдвиг всех ее столбцов, начиная со второго, так, чтобы максимальный элемент каждого столбца оказался в той же строке, что и максимальный элемент первого столбца.
46. Дана матрица  $A[m,n]$ . Поменять местами каждые две рядом расположенные строки, если максимальные элементы в них имеют одинаковую четность.
47. Дана матрица  $A[m,n]$ . Если минимальный элемент строки матрицы четный, выполнить циклический сдвиг этой строки вправо, чтобы этот элемент оказался в последнем столбце. Иначе выполнить циклический сдвиг строки влево, чтобы этот элемент оказался в первом столбце.

48. Дана целочисленная прямоугольная матрица. Определить номер строки матрицы, которая содержит максимальное количество четных элементов.
49. Понятие статической и динамической информации.
50. Фазы управления памятью.
51. Статическое управление памятью.
52. Стековое управление памятью
53. Управление кучей.
54. Понятие коллекций.
55. Виды коллекций.
56. Понятие *перечислителя*.
57. Класс `ArrayList`, основные операции над классом.
58. Понятие список, основные операции над списком.
59. Понятие очередь, основные операции над очередью.
60. Понятие стэк, основные операции над стэком
61. Описание и использование собственных функций Передача параметров и возврат результата работы функции.
62. Передача параметров по ссылке и по значению. Выходные параметры функций и их использование.
63. Понятие рекурсии.
64. Понятие векторной и растровой графики.
65. Связь C# со средой .NET Framework.
66. Понятие управляемый и неуправляемый код.
67. Понятие «небезопасный код».
68. Понятие указатель, объявление указателя, операторы \* и & в указателях.

## 9.2. Оценочные средства промежуточной аттестации в форме зачёта

Учебным планом зачёт по дисциплине не предусмотрен.

## 9.3. Оценочные средства для проведения промежуточной аттестации в форме курсовой работы

Учебным планом по дисциплине предусмотрена курсовая работа.

1. Шкала оценивания курсовой работы – «отлично», «хорошо», «удовлетворительно», «неудовлетворительно».
2. Тема курсовых работ унифицирована для всех обучающихся – «Разработка программного средства с графическим интерфейсом для манипулирования структурой данных» (по вариантам).  
Возможная тематика курсовых работ:  
Разработка графического интерфейса для работы со списками.  
Разработка графического интерфейса для работы с деревьями и графами.  
Разработка графического интерфейса для создания и отображения сложных пользовательских объектов.

3. Критерии итоговой оценки за курсовую работу приведены в таблице 5.

Таблица 5. Оцениваемые показатели для проведения промежуточной аттестации в форме курсовой работы

№ раздела	Наименование раздела	Баллы по шкале уровня
1	Введение	Выше базового – 2 Базовый – 1 Ниже базового – 0
2	Постановка задачи	Выше базового – 2 Базовый – 1 Ниже базового – 0
3	Теоретическое описание	Выше базового – 2 Базовый – 1 Ниже базового – 0
4	Программное средство	Выше базового – 2 Базовый – 1 Ниже базового – 0
5	Библиографический список	Выше базового – 2 Базовый – 1 Ниже базового – 0

Критерии итоговой оценки за курсовую работу:

«отлично» – при сумме баллов от 8 до 10;

«хорошо» – при сумме баллов от 5 до 7;

«удовлетворительно» – при сумме баллов от 3 до 5;

«неудовлетворительно» – при сумме баллов менее 3 .

4. Курсовая работа состоит из титульного листа, общей части, специальной части и приложений.

Общая часть включает в себя задание на проектирование, результаты анализа объектов предметной области.

В специальной части описаны используемые в программе структур данных, алгоритм всех функций программы в виде псевдокода, обоснование и выбор программных решений.

В приложение необходимо представить исходные тексты программ и внешний вид проектируемой системы.

Дополнительные процедурные сведения:

а) требования к срокам выполнения этапов работы и представления её окончательного варианта руководителю содержатся в методических указаниях;

б) проверку и оценку работы осуществляет руководитель, который доводит до сведения обучающегося достоинства и недостатки работы и его оценку. Оценка проставляется в зачётную книжку обучающегося и ведомость для курсовых работ. Если обучающийся не согласен с оценкой руководителя, проводится защита работы перед комиссией, которую назначает заведующий кафедрой;

в) работа не подлежит обязательному рецензированию.

В процессе выполнения курсовой работы руководитель осуществляет систематическое консультирование.

5. Дополнительные процедурные сведения:

– студенты выбирают тему для курсовой работы самостоятельно из предложенного списка и согласовывают свой выбор с преподавателем в течение двух первых недель обучения;

– проверку и оценку работы осуществляет руководитель, который доводит до сведения обучающегося достоинства и недостатки курсовой работы и ее оценку. Оценка проставляется в зачетную книжку обучающегося и ведомость для курсовой работы. Если обучающийся не согласен с оценкой руководителя, проводится защита работы перед комиссией, которую назначает заведующий кафедрой;

– защита курсовой работы проводится в течение двух последних недель семестра и выполняется в форме устной защиты в виде доклада и презентации на 5-7 минут с последующим ответом на поставленные вопросы, в ходе которых выясняется глубина знаний студента и самостоятельность выполнения работы;

– работа не подлежит обязательному внешнему рецензированию;

– курсовые работы хранятся на кафедре в течение трех лет.

## **10. Методические рекомендации по организации изучения дисциплины**

Студенты перед началом изучения дисциплины ознакомлены с системами кредитных единиц и балльно-рейтинговой оценки.

Студенты, изучающие дисциплину, обеспечиваются электронными изданиями или доступом к ним, учебно-методическим комплексом по дисциплине.

## **11. Внесение изменений и дополнений в рабочую программу дисциплины**

Содержание рабочих программ дисциплин ежегодно обновляется протоколами заседаний кафедры по утвержденной «Положением о структуре, содержании и оформлении рабочих программ дисциплин по образовательным программам, соответствующим ФГОС ВО с учетом профессиональных стандартов» форме.

## Приложение 1

Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
**«Тверской государственный технический университет»**

Направление подготовки бакалавров 09.03.01 Информатика  
и вычислительная техника

Направленность (профиль): Вычислительные машины, комплексы, системы и сети

Кафедра «Электронные вычислительные машины»

Дисциплина «Алгоритмические языки и программирование»

Семестр 1

### ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 1

*1. Вопрос для проверки уровня «ЗНАТЬ» — 0 или 2 балла:*

Переменные в C#. Имена переменных и их значения. Числовые литералы (константы). Символьные и строковые литералы (константы).

*2. Вопрос для проверки уровня «УМЕТЬ» — 0 или 1 балл:*

Обработка массивов: создание одного массива из другого.

*3. Задание для проверки уровня «УМЕТЬ» — 0 или 2 балла:*

Обработка массивов: поиск одинаковых и различных элементов массива.

**Критерии итоговой оценки за экзамен:**

«отлично» — при сумме баллов 5 или 6;

«хорошо» — при сумме баллов 4;

«удовлетворительно» — при сумме баллов 3;

«неудовлетворительно» — при сумме баллов 0, 1 или 2.

Составитель: ст. преподаватель кафедры ЭВМ \_\_\_\_\_ А.Н. Неведомский

Заведующий кафедрой ЭВМ \_\_\_\_\_ А.Р. Хабаров

## Приложение 2

Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Тверской государственный технический университет»

Направление подготовки бакалавров 09.03.01 Информатика  
и вычислительная техника

Направленность (профиль): Вычислительные машины, комплексы, системы и сети

Кафедра «Электронные вычислительные машины»

Дисциплина «Алгоритмические языки и программирование»

Семестр 2

### ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 1

*1. Вопрос для проверки уровня «ЗНАТЬ» — 0 или 2 балла:*

Понятие статической и динамической информации.

*2. Вопрос для проверки уровня «УМЕТЬ» — 0 или 1 балл:*

Статическое управление памятью.

*3. Задание для проверки уровня «УМЕТЬ» — 0 или 2 балла:*

Показать пример применения векторной и растровой графики

**Критерии итоговой оценки за экзамен:**

«отлично» — при сумме баллов 5 или 6;

«хорошо» — при сумме баллов 4;

«удовлетворительно» — при сумме баллов 3;

«неудовлетворительно» — при сумме баллов 0, 1 или 2.

Составитель: ст. преподаватель кафедры ЭВМ \_\_\_\_\_ А.Н. Неведомский

Заведующий кафедрой ЭВМ \_\_\_\_\_ А.Р. Хабаров