

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

УТВЕРЖДАЮ

Проректор
по учебной работе

_____ Э.Ю. Майкова
« ____ » _____ 20__ г.

РАБОЧАЯ ПРОГРАММА
обязательной части Блока 1 «Дисциплины (модули)»
«Алгоритмизация и программирование»

Направление подготовки бакалавров – 09.03.03 Прикладная информатика
Направленность (профиль) – Прикладная информатика в экономике
Типы задач профессиональной деятельности – проектный, организационно-
управленческий

Форма обучения – очная и заочная

Факультет информационных технологий
Кафедра «Информационные системы»

Тверь 20

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

Разработчик программы:
Ст. преп. кафедры ИС

Ю.Г. Козлова

Программа рассмотрена и одобрена на заседании кафедры ИС
«_____» _____ 20 ____ г., протокол № ____.

Заведующий кафедрой ИС

Б.В. Палюх

Согласовано
Начальник учебно-методического
отдела УМУ

Д.А. Барчуков

Начальник отдела
комплектования
зональной научной библиотеки

О.Ф. Жмыхова

1. Цели и задачи дисциплины

Целью дисциплины «Алгоритмизация и программирование» является формирование базовых знаний и навыков в области основ алгоритмизации и программирования для решения типичных задач, путем разработки и отладки программ в современных средах программирования.

Задачами изучения дисциплины являются:

- формирование у обучающихся алгоритмического мышления, систематизация принципов построения языков программирования и подходов к разработке программ для ЭВМ;

- формирование навыков реализации алгоритмов на высокоуровневом императивном языке программирования; разработки, отладки и тестирования программ;

- подготовка обучающихся к системному восприятию дальнейших дисциплин из учебного плана, использующих навыки алгоритмизации и программирования;

- получение представлений об основных идеях структурного и объектно-ориентированного программирования и развитие способностей сознательно использовать материал курса, умение разбираться в существующих языковых и программных средствах и условиях их применения.

2. Место дисциплины в образовательной программе

Дисциплина относится к дисциплинам обязательной части Блока 1 ОП ВО. Для изучения курса требуются знания, умения и навыки, полученные в ходе изучения дисциплин «Операционные системы», «Вычислительные системы, сети и телекоммуникации».

Приобретенные знания и умения в рамках данной дисциплины необходимы в дальнейшем при изучении дисциплины «Программная инженерия», «Проектирование информационных систем», прохождении практик и при выполнении выпускной квалификационной работы.

3. Планируемые результаты обучения по дисциплине

3.1 Планируемые результаты обучения по дисциплине

Компетенция, закрепленная за дисциплиной в ОХОП:

УК-4. Способен осуществлять деловую коммуникацию в устной и письменной формах на государственном языке Российской Федерации и иностранном(ых) языке(ах)

Индикаторы компетенции, закреплённые за дисциплиной в ОХОП:

ИУК-4.3. Использует современные информационно-коммуникативные технологии.

Показатели оценивания индикаторов достижения компетенций:

Знать:

З1. современные информационные технологии и программные средства, используемые для описания и декомпозиции поставленных задач профессиональной деятельности;

З2. современные информационные технологии и программные средства, используемые разработки алгоритмов и блок-схем.

З3. современные интегрированные среды для разработки программ, в том числе отечественного производства при решении задач профессиональной деятельности;

Уметь:

У1. на основе поставленной задачи описать и провести декомпозицию поставленных задач профессиональной деятельности;

У2. на основе поставленной задачи, разработать или применить известный алгоритм с использованием программных средств, используемых для разработки алгоритмов и блок-схем;

У3. на основе поставленной задачи, разработать программу на языке высокого уровня с использованием интегрированных сред для разработки;

Компетенция, закреплённая за дисциплиной в ОХОП:

ОПК-7. Способен разрабатывать алгоритмы и программы, пригодные для практического применения

Индикаторы компетенции, закреплённые за дисциплиной в ОХОП:

ИОПК-7.1. Применяет языки программирования, работает с базами данных, современными программными средами разработки информационных систем и технологий для автоматизации бизнес-процессов, решения прикладных задач различных классов, ведения баз данных и информационных хранилищ.

Показатели оценивания индикаторов достижения компетенций:

Знать:

З1. Основные структуры данных и методы их обработки;

З2. Основные алгоритмы для сортировки, поиска и обработки информации;

З3. Различия между языками программирования высокого и низкого уровня;

З4. Основные принципы и понятия объектно-ориентированного подхода.

Уметь:

У1. Формализовать прикладную задачу, выбирать для неё подходящие структуры данных и алгоритмы обработки.

У2. Программировать алгоритмы, используя средства языков высокого уровня и объектно-ориентированного подхода.

У3. Разрабатывать тестовые случаи и сценарии.

3.2. Технологии, обеспечивающие формирование компетенций

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

4. Трудоемкость дисциплины и виды учебной работы ОЧНАЯ ФОРМА ОБУЧЕНИЯ

Таблица 1а. Распределение трудоемкости дисциплины по видам учебной работы

Вид учебной работы	Зачетные единицы	Академические часы
Общая трудоемкость дисциплины	8	288
Аудиторные занятия (всего)		135
В том числе:		
Лекции		45

Практические занятия (ПЗ)		не предусмотрены
Лабораторные работы (ЛР)		90
Самостоятельная работа обучающихся (всего)		117+36
В том числе:		
Курсовая работа		30
Курсовой проект		не предусмотрен
Расчетно-графические работы		не предусмотрены
Реферат		не предусмотрен
Другие виды самостоятельной работы:		
- изучение теоретической части дисциплины;		30
- подготовка к защите лабораторных работ		45
Текущий контроль успеваемости и промежуточная аттестация (зачет)		12
Текущий контроль успеваемости и промежуточная аттестация (экзамен)		36 (экз)
Практическая подготовка при реализации дисциплины (всего)		0

ЗАОЧНАЯ ФОРМА ОБУЧЕНИЯ

Таблица 1б. Распределение трудоемкости дисциплины по видам учебной работы

Вид учебной работы	Зачетные единицы	Академические часы
Общая трудоемкость дисциплины	8	288
Аудиторные занятия (всего)		20
В том числе:		
Лекции		8
Практические занятия (ПЗ)		не предусмотрены
Лабораторные работы (ЛР)		12
Самостоятельная работа обучающихся (всего)		255+13
В том числе:		
Курсовая работа		20
Курсовой проект		не предусмотрен
Расчетно-графические работы		не предусмотрены
Реферат		не предусмотрен
Другие виды самостоятельной работы:		
- изучение теоретической части дисциплины;		110
- подготовка к защите лабораторных работ		90
Текущий контроль успеваемости и промежуточная аттестация (зачет)		12
Текущий контроль успеваемости и промежуточная аттестация (экзамен)		36 (экз)
Практическая подготовка при реализации дисциплины (всего)		0

5. Структура и содержание дисциплины

5.1. Структура дисциплины

ОЧНАЯ ФОРМА ОБУЧЕНИЯ

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

№	Наименование модуля	Труд-ть часы	Лекции	Практич. занятия	Лаб. работы	Сам. работа
1	Основы алгоритмизации.	33	2	-	6	10+2
2	Общая характеристика языка C++. Синтаксис языка C++. Управляющие конструкции языка C++.	40	4	-	15	15+5
3	Массивы, структуры и объединения	33	3	-	10	10+3
4	Указатели. Функции. Динамические структуры данных.	33	6	-	14	13+5
5	Основы объектно-ориентированного программирования на языке C++	59	16	-	20	23+7
6	Принципы разработки объектно-ориентированных программ в среде разработчика Microsoft Visual C++	57	10	-	19	23+7
7	Многопоточное программирование.	33	4	-	6	23+7
Всего на дисциплину		288	45	-	90	117+36

ЗАОЧНАЯ ФОРМА ОБУЧЕНИЯ

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

№	Наименование модуля	Труд-ть часы	Лекции	Практич. занятия	Лаб. работы	Сам. работа
1	Основы алгоритмизации.	33	1	-	1	35+1
2	Общая характеристика языка C++. Синтаксис языка C++. Управляющие конструкции языка C++.	40	1	-	2	35+2
3	Массивы, структуры и объединения	33	1	-	1	35+2
4	Указатели. Функции. Динамические структуры данных.	33	1	-	2	35+2
5	Основы объектно-ориентированного программирования на языке C++	59	2	-	2	45+2
6	Принципы разработки объектно-ориентированных программ в среде разработчика Microsoft Visual C++	57	1	-	2	35+2
7	Многопоточное программирование.	33	1	-	2	35+2
Всего на дисциплину		288	8	-	12	255+13

5.2. Содержание дисциплины

Модуль 1 Основы алгоритмизации

Базовые алгоритмические структуры: следование, ветвление, повторение. Алгоритмические языки программирования. Алфавит, синтаксис и семантика алгоритмических языков программирования. Инструменты и схема построения исполняемого модуля для программы на языке программирования высокого уровня.

Интегрированные среды разработки. Основы алгоритмизации. Фон-неймановские принципы работы компьютеров. Алгоритм. Свойства алгоритма: понятность, дискретность, детерминированность, конечность, результативность, массовость. Основные этапы разработки алгоритмов: постановка задачи, построение математической модели, разработка алгоритма решения задачи, проверка правильности и оценка сложности алгоритма. Формы записи алгоритмов: словесно-пошаговая, графическая, в псевдокодах, на алгоритмических языках.

Модуль 2. Общая характеристика языка C++. Синтаксис языка C++. Управляющие конструкции языка C++.

Общая характеристика языка C++. История и условия возникновения. Базовые принципы и свойства языка C++. Простейшая программа.

Синтаксис языка C++. Алфавит, идентификаторы, ключевые слова. Комментарии. Типы, константы. Операции и выражения. Приоритет операций. Общая структура программы.

Управляющие конструкции языка C++. Условный оператор. Оператор ветвления. Операторы циклов. Прерывание циклов. Инвариант цикла.

Модуль 3. Массивы, структуры и объединения

Массивы. Многомерные массивы. Массивы как параметры функций. Типичные операции над массивами. Алгоритмы сортировки массивов и поиска в массиве.

Структуры и функции. Массивы структур. Доступ к элементам структуры. Объединения. Битовые поля структур и объединений. Указатели на структуры. Структуры со ссылками на себя. Средство typedef. Объединения. Битовые поля. Размещение структур в памяти

Модуль 4. Указатели. Функции. Динамические структуры данных.

Указатели. Связь указателей и массивов. Динамические массивы. Указатели на функции. Указатели и параметры функций. Сложные описания с указателями. Операции индексирования, взятия адреса и разадресации. Функции пользователя: Передача параметров по значению, по адресу, по ссылке. Параметры по умолчанию. Передача функции в качестве параметра. Прототипирование. Динамическое распределение памяти, операторы new и delete. Динамические структуры данных: стеки, очереди, списки, бинарные деревья.

Модуль 5. Основы объектно-ориентированного программирования на языке C++

Объект, класс, данные, методы, доступ; проектирование объектно-ориентированных программ: методы и алгоритмы. Конструкторы и деструкторы. Встраиваемые и перегружаемые функции.

Наследование свойств; системы объектов и классов. Контроль доступа при наследовании.

Дружественные функции и классы. Полиморфизм. Виртуальные функции. Определение виртуальных функций. Статическое и динамическое связывание.

Перегрузка операций. Перегрузка операции присваивания. Абстрактные и виртуальные классы. Статические элементы классов. Доступ к статическим элементам.

Потоковый ввод-вывод. Базовые объекты ввода – вывода. Форматирование: функции форматирования, флаги форматирования, манипуляторы, ошибки потоков. Обращение к файлам как к потокам.

Шаблоны классов и функций. Обработка исключительных ситуаций. Операторы catch, try, throw.

Модуль 6 Принципы разработки объектно-ориентированных программ в среде разработчика Microsoft Visual C++

Разработка приложения на основе интерфейса SDI и CLI.

Разработка проектов на основе классов MFC. Типы мастеров проекта. Изменение установок проекта. Иерархия классов MFC. Функции MFC. Обработка сообщений в MFC. Анализ классов проекта. Создание ресурсов. Включение инструмента ClassWizard. Типы диалогов. Введение элементов в диалоговое окно. Виды применяемых элементов. Управление мышью. Поддержка вывода на принтер. Запись в файл.

Разработка проектов на основе функций Win API. Виды интерфейсов API. Принципы разработки проекта на основе Win API. Типы мастеров проекта. Технологии разработки консольных и оконных интерфейсов. Использование в проекте DLL.

Модуль 7 Многопоточное программирование

Создание потоков. Рабочие и интерфейсные потоки. Синхронное и асинхронное программирование. Семафоры и мьютексы. Виды мьютексов. Синхронизация потоков. Остановка потоков.

5.3. Лабораторные работы

ОЧНАЯ ФОРМА ОБУЧЕНИЯ

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

Модули. Цели ЛР	Наименование лабораторных работ	Трудоем- кость в часах
Модуль 1 Цель: получение навыков разработки алгоритмов для решения различных	Основные этапы разработки алгоритмов. Применение различных форм записи алгоритмов	2

классов задач	Построение блок-схем	2
Модуль 2 Цель: научиться реализовывать на языке C++ алгоритмы линейной структуры на языке, ветвления и циклы	Изучение среды Visual Studio. Создание простейших проектов.	2
	Средства отладки и тестирования Visual Studio.	2
	Линейные процессы в C++	2
	Ветвления в C++	5
	Циклы в C++.	5
Модуль 3 Цель: научиться работать с массивами, структурами и объединениями, уметь реализовывать поиск в них и осуществлять сортировки с помощью различных алгоритмов	Одномерные и многомерные массивы	2
	Алгоритмы поиска	2
	Алгоритмы сортировки	2
	Структуры	2
	Объединения	2
Модуль 4 Цель: научиться работать с указателями, функциями и динамическими структурами данных	Указатели	2
	Функции	4
	Рекурсия	2
	Динамические структуры данных	6
Модуль 5 Цель: научиться на практике применять принципы объектно-ориентированного программирования.	Создание простейших классов	4
	Работа с конструкторами и деструкторами	4
	Наследование	4
	Дружественные функции и классы	4
	Шаблоны классов и функций	4
Модуль 6 Цель: научиться разрабатывать визуальные приложения на основе интерфейса CLI	Создание графического интерфейса в Visual Studio	4
	Разработка алгоритма оконного приложения.	4
	Создание интерфейса приложения	6
	Создание процедур обработки событий.	6
Модуль 7 Цель: научиться работать с потоками в C++.	Создание многопоточного приложения	3
	Обмен данными между потоками.	3
	Остановка и запуск отдельного потока. Передача управления.	3

ЗАОЧНАЯ ФОРМА ОБУЧЕНИЯ

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

Модули. Цели ЛР	Наименование лабораторных работ	Трудоем- кость в часах
Модуль 1 Цель: получение навыков разработки алгоритмов для решения различных классов задач	Основные этапы разработки алгоритмов. Применение различных форм записи алгоритмов	0,5
	Построение блок-схем	0,5
Модуль 2 Цель: научиться реализовывать на	Изучение среды Visual Studio. Создание простейших проектов.	0,5

языке С++ алгоритмы линейной структуры на языке, ветвления и циклы	Линейные процессы, ветвления и циклы в С++	0,75
Модуль 3 Цель: научиться работать с массивами, структурами и объединениями, уметь реализовывать поиск в них и осуществлять сортировки с помощью различных алгоритмов	Одномерные и многомерные массивы. Алгоритмы поиска и сортировки.	0,75
	Структуры и объединения	0,5
Модуль 4 Цель: научиться работать с указателями, функциями и динамическими структурами данных	Указатели. Функции. Рекурсия	0,75
	Динамические структуры данных	0,75
Модуль 5 Цель: научиться на практике применять принципы объектно-ориентированного программирования.	Создание простейших классов	0,5
	Работа с конструкторами и деструкторами	1
	Наследование	0,5
	Дружественные функции и классы	0,5
Модуль 6 Цель: научиться разрабатывать визуальные приложения на основе интерфейса СLI	Шаблоны классов и функций	0,5
	Создание графического интерфейса в Visual Studio. Разработка алгоритма оконного приложения	1
Модуль 7 Цель: научиться работать с потоками в С++.	Создание интерфейса приложения и процедур обработки событий.	1
	Создание многопоточного приложения	1
	Обмен данными между потоками. Остановка и запуск отдельного потока. Передача управления.	1

5.4. Практические занятия

Учебным планом лабораторные работы не предусмотрены.

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 изучении отдельных тем курса по заданию преподавателя по рекомендуемой им учебной литературе, в подготовке к лабораторным занятиям, к текущему контролю успеваемости, экзамену в выполнении курсовой работы.

После вводных занятий, в которых обозначается содержание дисциплины, ее проблематика и практическая значимость, студентам выдается задание на курсовое проектирование.

Курсовая работа выполняется в соответствии с методическими указаниями по выполнению курсовой работы, разработанными на кафедре.

В рамках дисциплины выполняется 28 лабораторных работ, которые защищаются устным опросом. Выполнение всех лабораторных работ обязательно.

В случае невыполнения лабораторной работы по уважительной причине студент должен выполнить пропущенные лабораторные занятия в часы, отведенные на консультирование с преподавателем.

7. Учебно-методическое и информационное обеспечение дисциплины

7.1. Основная литература по дисциплине

1. Трофимов, В.В. Алгоритмизация и программирование : учебник для вузов : в составе учебно-методического комплекса / В.В. Трофимов, Т.А. Павловская; под редакцией В.В. Трофимова. - Москва : Юрайт, 2023. - (Высшее образование) (УМК-У). - Образовательная платформа Юрайт. - Режим доступа: по подписке. - Дата обращения: 12.09.2022. - Текст : электронный. - ISBN 978-5-534-07834-3. - URL: <https://urait.ru/bcode/513269> . - (ID=145176-0)

2. Огнева, М. В. Программирование на языке С++: практический курс : учебное пособие для вузов / М. В. Огнева, Е. В. Кудрина. — Москва : Издательство Юрайт, 2024. — 335 с. — (Высшее образование). — ISBN 978-5-534-05123-0. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/539713> (дата обращения: 29.03.2024). - (ID=136081-0)

3. Андрианова, А.А. Алгоритмизация и программирование. Практикум : учебное пособие / А.А. Андрианова, Л.Н. Исмагилов, Т.М. Мухтарова. - Санкт-Петербург [и др.] : Лань, 2022. - ЭБС Лань. - Текст : электронный. - Режим доступа: по подписке. - Дата обращения: 07.07.2022. - ISBN 978-5-8114-3336-0. - URL: <https://e.lanbook.com/book/206258> . - (ID=146316-0)

4. Конова, Е.А. Алгоритмы и программы. Язык С++ : учебное пособие по направлению "Прикладная информатика" / Е.А. Конова, Г.А. Поллак. - 6-е изд. - Санкт-Петербург [и др.] : Лань, 2021. - ЭБС Лань. - Текст : электронный. - Режим доступа: по подписке. - Дата обращения: 07.07.2022. - ISBN 978-5-8114-8487-4 . - URL: <https://e.lanbook.com/book/176900> . - (ID=113797-0)

7.2. Дополнительная литература

1. Страустрап, Б. Введение в язык С++. Турне по С ++ / Б. Страустрап. - [Б. м.] : [б. и.], 2011. - Внешний сервер. - Текст : электронный. - URL: <http://citforum.ru/programming/cpp/aglav.shtml> . - (ID=86034-0)

2. Юрина, Т.А. Программирование и алгоритмизация : учебно-методическое пособие / Т.А. Юрина; Сибирский государственный автомобильно-дорожный университет. - Омск : Сибирский государственный автомобильно-дорожный университет, 2021. - ЭБС Лань. - Текст : электронный. - Режим доступа: по подписке. - Дата обращения: 07.07.2022. - URL: <https://e.lanbook.com/book/179228> . - (ID=147223-0)

3. Долгов, А.И. Алгоритмизация прикладных задач : учебное пособие / А.И. Долгов. - 3-е изд. ; стер. - Москва : Флинта, 2021. - ЭБС Университетская библиотека онлайн. - Текст : электронный. - Дата обращения: 07.07.2022. - Режим доступа: по подписке. - ISBN 978-5-9765-0086-2. - URL: https://biblioclub.ru/index.php?page=book_view_red&book_id=83142 . - (ID=113601-0)

4. Липанова, И.А. Алгоритмизация и программирование : учебно-методическое пособие по выполнению курсовых работ / И.А. Липанова, В.А. Медведев, С.В. Хорошенко; Санкт-Петербургский государственный университет телекоммуникаций имени М.А. Бонч-Бруевича. - Санкт-Петербург : Санкт-Петербургский государственный университет телекоммуникаций имени М.А. Бонч-Бруевича, 2015. - ЭБС Лань. - Текст : электронный. - Режим доступа: по подписке. - Дата обращения: 07.07.2022. - URL: <https://e.lanbook.com/book/180026> . - (ID=147630-0)

5. Кудрина, Е. В. Основы алгоритмизации и программирования на языке C# : учебное пособие для вузов / Е. В. Кудрина, М. В. Огнева. — Москва : Издательство Юрайт, 2024. — 322 с. — (Высшее образование). — ISBN 978-5-534-09796-2. — Текст : электронный // Образовательная платформа Юрайт [сайт]. — URL: <https://urait.ru/bcode/541687> (дата обращения: 29.03.2024). - (ID=145304-0)

7.3. Методические материалы

1. Учебно-методический комплекс дисциплины обязательной части Блока 1 "Алгоритмизация и программирование". Направление подготовки 09.03.03 Прикладная информатика. Направленность (профиль): Прикладная информатика в экономике : ФГОС 3++ / Каф. Информационных систем ; сост. Ю.Г. Козлова. - 2022. - (УМК). - Текст : электронный. - URL: <https://elib.tstu.tver.ru/MegaPro/GetDoc/Megapro/159432> . - (ID=159432-0)

2. Виноградов, Г.П. Алгоритмизация и программирование на языке VBA. Примеры и задачи : в составе учебно-методического комплекса / Г.П. Виноградов; Тверской государственный технический университет, Кафедра ИПМ. - Тверь : ТвГТУ, 2012. - (УМК-М). - Текст : электронный. - (ID=123467-0)

3. 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: <https://elib.tstu.tver.ru/MegaPro/GetDoc/Megapro/61198> . - (ID=61198-1)

4. 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. - 156 с. : ил. - Библиогр. : с. 153. - Текст : непосредственный. - ISBN 5-7995-0342-2 : 137 p. - (ID=60503-82)

7.4. Программное обеспечение по дисциплине

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

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

Microsoft Visual Studio.

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/159432>

8. Материально-техническое обеспечение дисциплины

При изучении дисциплины «Алгоритмизация и программирование» используются современные средства обучения: наглядные пособия, диаграммы, схемы.

Возможна демонстрация лекционного материала с помощью оверхед-проектора (кодоскопа) и мультипроектора.

9. Оценочные средства для проведения промежуточной аттестации

9.1. Оценочные средства для проведения промежуточной аттестации в форме экзамена

1. Экзаменационный билет соответствует форме, утвержденной Положением о рабочих программах дисциплин, соответствующих федеральным государственным образовательным стандартам высшего образования с учетом профессиональных стандартов. 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.

Число экзаменационных билетов – 20. Число вопросов (заданий) в экзаменационном билете – 3 (1 вопрос для категории «знать» и 2 вопроса для категории «уметь»).

Продолжительность экзамена – 60 минут.

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

3. Критерии оценки за экзамен:

для категории «знать»:

выше базового – 2;

базовый – 1;

ниже базового – 0;

критерии оценки и ее значение для категории «уметь»:

отсутствие умения – 0 балл;

наличие умения – 2 балла.

«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.

4. Вид экзамена – письменный экзамен, включающий решение задач с использованием ЭВМ.

5. База заданий, предъявляемая обучающимся на экзамене

1. Факторы, обусловившие появление и содержание концепции ООП.

2. Основные свойства ООП: абстракция, инкапсуляция, наследование, полиморфизм.

3. Проектирование объектно-ориентированных программ: методы и алгоритмы.

4. Конструкторы и деструкторы.

5. Встраиваемые и перегружаемые функции.

6. Наследование свойств; системы объектов и классов. Контроль доступа при наследовании.

7. Дружественные функции и классы.

8. Полиморфизм.

9. Виртуальные функции. Определение виртуальных функций. Статическое и динамическое связывание.

10. Перегрузка операций. Перегрузка операции присваивания.

11. Абстрактные и виртуальные классы.

12. Статические элементы классов. Доступ к статическим элементам.

13. Поточковый ввод-вывод. Базовые объекты ввода – вывода. Форматирование: функции форматирования, флаги форматирования, манипуляторы, ошибки потоков. Обращение к файлам как к потокам.

14. Шаблоны классов и функций.

15. Обработка исключительных ситуаций. Операторы catch, try, throw.

16. Разработка приложения на основе интерфейса SDI и CLI.

17. Разработка проектов на основе классов MFC.

18. Разработка проектов на основе функций Win API.

19. Виды интерфейсов API.

20. Технологии разработки консольных и оконных интерфейсов.

21. Многопоточное программирование. Создание потоков. Рабочие и интерфейсные потоки.

22. Синхронное и асинхронное программирование.

23. Семафоры и мьютексы. Виды мьютексов.

24. Синхронизация потоков. Остановка потоков.

Пользование различными техническими устройствами, кроме ЭВМ компьютерного класса и программным обеспечением, необходимым для решения поставленных задач, не допускается. При желании студента покинуть пределы аудитории во время экзамена экзаменационный билет после его возвращения заменяется.

Преподаватель имеет право после проверки письменных ответов на экзаменационные вопросы и решенных на компьютере задач задавать студенту в устной форме уточняющие вопросы в рамках содержания экзаменационного билета, выданного студенту.

Иные нормы, регламентирующие процедуру проведения экзамена, представлены в Положении о текущем контроле успеваемости и промежуточной аттестации студентов.

9.2. Оценочные средства для проведения промежуточной аттестации в форме зачета

1. Вид промежуточной аттестации в форме зачета.

Вид промежуточной аттестации устанавливается преподавателем:

по результатам текущего контроля знаний и умений обучающегося без дополнительных контрольных испытаний;

по результатам выполнения дополнительного итогового контрольного испытания при наличии у студентов задолженностей по текущему контролю.

2. При промежуточной аттестации без выполнения дополнительного итогового контрольного испытания студенту в обязательном порядке описываются критерии проставления зачёта:

«зачтено» - выставляется обучающемуся при условии выполнения им всех контрольных мероприятий: посещение лекций в объеме не менее 80% контактной работы с преподавателем, выполнения и защиты практических работ.

При промежуточной аттестации с выполнением заданий дополнительного итогового контрольного испытания студенту выдается билет с вопросами и задачами.

Число заданий для дополнительного итогового контрольного испытания - 10.

Число вопросов – 3 (2 вопроса для категории «знать» и 1 вопрос для категории «уметь»).

Продолжительность – 60 минут.

3. Шкала оценивания промежуточной аттестации – «зачтено», «не зачтено».

4. Критерии выполнения контрольного испытания и условия проставления зачёта:

для категории «знать» (бинарный критерий):

ниже базового - 0 балл;

базовый уровень – 1 балла;

критерии оценки и ее значение для категории «уметь» (бинарный критерий):

отсутствие умения – 0 балл;

наличие умения – 1 балла.

Критерии итоговой оценки за зачет:

«зачтено» - при сумме баллов 2 или 3;

«не зачтено» - при сумме баллов 0 или 1.

5. Для дополнительного итогового контрольного испытания студенту в обязательном порядке предоставляется:

база заданий, предназначенных для предъявления обучающемуся на дополнительном итоговом контрольном испытании (типовой образец задания приведен в Приложении);

методические материалы, определяющие процедуру проведения дополнительного итогового испытания и проставления зачёта.

6. Задание выполняется письменно и с использованием ЭВМ.

Перечень вопросов дополнительного итогового контрольного испытания:

1. Базовые алгоритмические структуры

2. Алгоритм. Свойства алгоритма. Основные этапы разработки алгоритмов. Формы записи алгоритмов: словесно-пошаговая, графическая, в псевдокодах, на алгоритмических языках.

3. Общая характеристика языка C++. История и условия возникновения. Базовые принципы и свойства языка C++.

4. Синтаксис языка C++. Алфавит, идентификаторы, ключевые слова. Комментарии. Типы, константы. Операции и выражения. Приоритет операций. Общая структура программы.

5. Операторы ветвления и множественного выбора.

6. Операторы циклов. Прерывание циклов. Инвариант цикла.

7. Массивы. Принципы размещения массивов в памяти. Обращение к элементам массива.

8. Многомерные массивы. Массивы как параметры функций. Типичные операции над массивами.

9. Общая характеристика алгоритмов сортировки данных.

10. Сортировка данных методом прямого включения.

11. Алгоритм сортировки данных методом прямого выбора.

12. Алгоритм сортировки данных методом прямого обмена.

13. Особенности шейкерной сортировки элементов массива.

14. Сортировка Шелла.

15. Сортировка элементов массива с помощью пирамиды.

16. Методы поиска в линейных неупорядоченных таблицах.

17. Понятия структуры и функции. Массивы структур. Доступ к элементам структуры.

18. Объединения. Битовые поля структур и объединений.

19. Указатели. Связь указателей и массивов. Сложные описания с указателями. Операции индексирования, взятия адреса и разадресации.

20. Динамические массивы.

21. Указатели на функции. Указатели и параметры функций.

22. Функции пользователя: Передача параметров по значению, по адресу, по ссылке.

23. Функции пользователя: Параметры по умолчанию. Передача функции в качестве параметра.

24. Прототипирование.

25. Динамическое распределение памяти, операторы new и delete.

26. Динамические структуры данных: стеки

27. Динамические структуры данных: очереди
28. Динамические структуры данных: списки
29. Динамические структуры данных: бинарные деревья.

9.3. Оценочные средства для проведения промежуточной аттестации в форме курсового проекта или курсовой работы

Учебным планом предусмотрена промежуточная аттестация в форме курсовой работы.

В процессе подготовки курсовой работы студент закрепляет и расширяет полученные знания по профессиональным и специальным дисциплинам и развивает необходимые способности и навыки самостоятельной исследовательской работы.

Курсовая работа имеет практическую направленность, в ходе ее выполнения студенты осуществляют поиск и исследование темы, демонстрируют навыки применения полученных знаний, методов и инструментов разработки программ для решения поставленной задачи.

Курсовая работа бакалавра по дисциплине «Алгоритмизация и программирование» должна содержать совокупность научно-прикладных результатов, свидетельствовать о способности автора уметь формулировать задачи разработки, самостоятельно вести поиск необходимой информации, используя теоретические знания и практические навыки. Она выполняется на базе теоретических знаний, умений и практических навыков, полученных студентом в предшествующий период обучения.

В ходе ее выполнения студенты осуществляют поиск информации, демонстрируют навыки применения полученных знаний, методов и инструментов разработки программ с помощью объектно-ориентированного подхода, и известных им алгоритмов.

Целью выполнения курсовой работы по дисциплине «Алгоритмизация и программирование» является применение полученных теоретических знаний и практических навыков для разработки и отладки эффективного алгоритма и программы с использованием современных технологий программирования для решения поставленной задачи.

Выполненная курсовая работа должна быть оформлена в соответствии с современными требованиями, привлечением современных средств редактирования и печати, представлена в виде рукописи (на бумажном носителе в твердом переплете) и в электронном виде.

Курсовая работа должна иметь логическую структуру и развернутый план, соответствующий теме. Курсовая работа в обязательном порядке состоит из титульного листа, оглавления, включающего в себя перечень структурных элементов курсовой работы с указанием номеров страниц, с которых начинается их местоположение в тексте, введения, основной части (содержит постановку задачи, описание метода решения, описание внешней спецификации, алгоритма решения задачи, результатов тестирования программы), заключения, списка использованных источников, и приложения по необходимости.

Рекомендуется иметь следующие структурные элементы в курсовой работе:

Титульный лист

Оглавление

Введение

1. Основная часть

основной части (содержит постановку задачи, описание метода решения, описание внешней спецификации, алгоритма решения задачи, результатов тестирования программы)

1.1. Постановка задачи

1.2. Описание метода решения

1.3. Описание внешней спецификации

1.4. Алгоритм решения задачи

1.5. Результаты тестирования программы

Заключение

Список литературы

Приложения

Общий объем курсовой работы не менее 20 листов печатного текста формата А4 (без учета *Приложений*).

Оглавление включает в себя перечень структурных элементов курсовой работы с указанием номеров страниц, с которых начинается их местоположение в тексте.

Введение должно содержать обоснование актуальности разрабатываемой темы, формулировку цели курсовой работы и перечень рассматриваемых задач.

Постановка задачи предполагает описание задачи, в котором строго определено: «что дано?», «каковы результаты?», «каковы ограничения на исходные данные?», «какова зависимость результата от исходных данных?»..

Описание метода решения описывает общий способ действий для достижения цели решаемой задачи. При разработке программы для описания методов решения используется функциональный подход.

Описание внешней спецификации содержит описание взаимодействия компоненты программы (программы, модуля, функции) с окружающим миром – с другими программами, человеком, внешним файлом и т.д. Внешняя спецификация несет двоякую функцию. Во-первых, она служит руководством по использованию программной компоненты, а во-вторых, она является формальным заданием для составления алгоритма программы.

Алгоритм решения задачи содержит описание алгоритма решаемой задачи в любой форме описания.

Заключение включает основные выводы по курсовой работе. В нем даётся оценка степени достижения цели и выполнения поставленных задач, полученных результатов, рекомендации по их практическому применению, а также определяются возможные перспективы развития в рамках предметной области.

Список литературы должен содержать библиографический перечень источников (включая и Интернет-ресурсы), информация из которых использовалась при выполнении курсовой работы.

Приложения включают дополнительную информацию по изучаемой предметной области, не вошедшую в основную часть.

Примерные темы курсовой работы:

1. Написать программу, которая формирует двумерную матрицу чисел размерностью $n \times n$. Затем формирует два одномерных массива, причем в один записываются элементы, расположенные на главной диагонали и выше, в другой – элементы матрицы, лежащие ниже главной диагонали и выводит все массивы на экран. Осуществить бинарный поиск заданного элемента во втором массиве. Результаты поиска вывести на экран.

2. Дана квадратная матрица размерностью $n \times n$. Написать программу формирования последовательности В, элементами которой являются элементы таблицы А, расположенные над главной диагональю. В полученной последовательности найти три наименьших элемента. Если количество элементов между 1-м и 2-м элементами совпадает с количеством элементов между 2-м и 3-м элементами, то поменять их местами, сохранив порядок следования (без использования дополнительного массива). Преобразованный массив вывести на экран.

3. Написать программу, которая предлагает пользователю некоторый список функций для построения графиков: $y=x^4/(x^3+1)$; $y=ax^2+bx+c$; $y=\sin(x)+b$; $y=1/(x^2+x+1)$. После выбора функции, задания коэффициентов и отрезка, на котором выполняется построение, программа строит соответствующий график. Затем значение коэффициентов и положение графика можно изменить (например, с помощью клавиш управления курсором), после чего график перестраивается и записывается обновленное уравнение кривой.

4. Написать программу, позволяющую отсортировать массив двумя способами (обменная и выбором). Обеспечить определение и отображение времени сортировки каждым методом. Предусмотреть операции создания файла, записи данных в файл и чтения файла для обработки.

5. Написать программу тестирования знаний студентов по дисциплине Программирование и основы алгоритмизации. Вопросы для тестирования должны выбираться из файла случайным образом. Программа должна осуществлять подсчет правильных ответов и выводить на экран результаты тестирования (общее число вопросов, количество правильных ответов, оценку).

6. Написать программу быстрой сортировки массива, двоичного поиска повторяющихся элементов и замены их пользователем. Предусмотреть операции создания файла, записи данных в файл и чтения файла для обработки.

По согласованию с руководителем работа может выполняться и по индивидуальной теме.

Шкала оценивания промежуточной аттестации в форме курсовой работы – «отлично», «хорошо», «удовлетворительно», «неудовлетворительно».

Оценка «отлично» выставляется в тех случаях, когда студент демонстрирует владение навыками самостоятельного решения поставленной проблемы, способность к содержательной презентации выполненной работы и защиты ее результатов при условии соответствия курсовой работы всем критериям в полном объеме.

Оценка «хорошо» ставится в случаях, когда студент демонстрирует высокий уровень владения проблемой исследования, способность к содержательной презентации выполненной работы и защиты ее результатов, но работа выполнена с незначительными отклонениями от требований.

Оценка «удовлетворительно» выставляется в тех случаях, когда студент хотя и демонстрирует владение проблемой исследования и представляет ее результаты, но при ответах допускает ошибочные утверждения, при этом в курсовой работе выявлены значительные отклонения от требований.

Оценка «неудовлетворительно» выставляется в ситуациях, когда обнаруживается несамостоятельность выполнения курсовой работы, некомпетентность в исследуемой студентом проблеме, при плохой защите курсовой работы, небрежном и неаккуратном ее оформлении

3. Критерии итоговой оценки за курсовую работу.

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	Описание метода решения	Выше базового – 2 Базовый – 1 Ниже базового – 0
4	Описание внешней спецификации	Выше базового – 2 Базовый – 1 Ниже базового – 0
5	Алгоритм решения задачи	Выше базового – 2 Базовый – 1 Ниже базового – 0
6	Результаты тестирования программы	Выше базового – 2 Базовый – 1 Ниже базового – 0

Критерии итоговой оценки за курсовую работу:

«отлично» – при сумме баллов от 10 до 12;

«хорошо» – при сумме баллов от 8 до 9;

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

«неудовлетворительно» – при сумме баллов менее 5, а также при любой другой сумме, если раздел «Алгоритм решения задачи» имеет 0 баллов.

10. Методические рекомендации по организации изучения дисциплины.

Студенты перед началом изучения дисциплины ознакомлены с системами кредитных единиц и балльно-рейтинговой оценки, которые должны быть опубликованы и размещены на сайте вуза или кафедры. Студенты, изучающие дисциплину обеспечиваются электронными изданиями или доступом к ним, учебно-методическим комплексом по дисциплине.

11. Внесение изменений и дополнений в рабочую программу дисциплины

Кафедра ежегодно обновляет содержание рабочих программ дисциплин, которые оформляются протоколами. Форма протокола утверждена Положением о структуре, содержании и оформлении рабочих программ дисциплин, по образовательным программам, соответствующих ФГОС ВО с учетом профессиональных стандартов

Федеральное государственное бюджетное образовательное учреждение
высшего образования

«Тверской государственный технический университет»

Направление подготовки бакалавров – 09.03.03 Прикладная информатика

Направленность (профиль) – Прикладная информатика в экономике

Кафедра «Информационных систем»

Дисциплина «Алгоритмизация и программирование»

Семестр 2

ЭКЗАМЕНАЦИОННЫЙ БИЛЕТ № 1

1. Вопрос для проверки уровня «ЗНАТЬ» – 0, 1 или 2 балла:

Операторы ветвления и множественного выбора в C++.

2. Задание для проверки уровня «УМЕТЬ» - 0, 1 или 2 балла:

Нарисовать блок-схему алгоритма пузырьковой сортировки.

3. Задание для проверки уровня «УМЕТЬ» – 0, 1 или 2 балла:

Сформировать квадратную матрицу размерностью $N \times M$ (количество строк и столбцов вводить с клавиатуры) по правилу: $A[i,j] = (i-j)/2$. Найти сумму элементов главной диагонали. Результат вывести на экран..

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 балла;

Составитель: ст. преп. _____ Ю.Г. Козлова

Заведующий кафедрой: д.т.н., профессор _____ Б.В. Палюх

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

Направление подготовки бакалавров – 09.03.03 Прикладная информатика
Направленность (профиль) – Прикладная информатика в экономике
Кафедра «Информационных систем»
Дисциплина «Алгоритмизация и программирование»
Семестр 1

ЗАДАНИЕ ДЛЯ ДОПОЛНИТЕЛЬНОГО ИТОГОВОГО КОНТРОЛЬНОГО ИСПЫТАНИЯ № 1

1. Вопрос для проверки уровня «ЗНАТЬ» – 0 или 1 балл:

Операторы ветвления и множественного выбора.

2. Вопрос для проверки уровня «ЗНАТЬ» – 0 или 1 балл:

Динамические структуры: списки.

3. Задание для проверки уровня «УМЕТЬ» – 0 или 1 балл:

Используя указатели, в массиве целых чисел из n элементов найти наиболее часто встречающееся число. Если таких чисел несколько, определить наименьшее из них.

Критерии итоговой оценки за зачет:

«зачтено» - при сумме баллов 2 или 3;

«не зачтено» - при сумме баллов 0 или 1.

Составитель: ст. преп. _____ Ю.Г. Козлова

Заведующий кафедрой: д.т.н., профессор _____ Б.В. Палюх