

Программа учебной дисциплины «Алгоритмизация и программирование»

Утверждена
Академическим советом ОП¹
Протокол № от __.__.20__

Разработчик	Сластников С.А., доцент ДПМ МИЭМ НИУ ВШЭ
Число кредитов	6
Контактная работа (час.)	130
Самостоятельная работа (час.)	98
Курс, Образовательная программа	1 курс, ОП «Прикладная математика»
Формат изучения дисциплины	Без использования онлайн курса

1. Цель, результаты освоения дисциплины и пререквизиты

Целями освоения дисциплины Алгоритмизация и программирование является ознакомление студентов с методами решения прикладных задач и их реализации на языке программирования Си. В процессе изучения дисциплины студенты приобретают навыки проектирования и разработки программных приложений, которые будут использоваться при выполнении различных заданий и работ по дисциплинам, изучаемым на последующих курсах.

В результате освоения дисциплины студент должен:

Знать:

Основы информатики.

Основные структуры данных и методы их обработки.

Различия между языками программирования высокого и низкого уровня.

Язык программирования Си.

Некоторые функции стандартной библиотеки языка Си.

¹ Для ПУД из общеуниверситетского пула – Руководитель Департамента.

Уметь:

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

Программировать алгоритмы, используя средства языка Си.

Разрабатывать тестовые случаи и сценарии.

Иметь навыки (приобрести опыт):

В разработке программ на языке Си.

В проведении отладки и тестирования разработанных программ.

В проведении анализа получаемых результатов и оформлении сопроводительной документации на программы.

Изучение данной дисциплины базируется на следующих дисциплинах:

- Линейная алгебра;
- Математический анализ;
- Дискретная математика.

Для освоения учебной дисциплины, студенты должны владеть следующими знаниями и компетенциями:

- знать основы информатики и ИКТ;
- знать основы линейной алгебры;
- знать элементы математической логики;
- уметь использовать программные и аппаратные средства информатики.

2. Содержание учебной дисциплины

Разработчик ПУД может совместить детализацию тематического содержания в таблице, или разместить темы (разделы дисциплины) отдельно, может избежать табличного отображения и привести разбиение на контактные часы и самостоятельную работу в виде описания.

Тема (раздел дисциплины)	Объем в часах ²	Планируемые результаты обучения (ПРО), подлежащие контролю	Формы контроля
	лк		
	см		
	онл/ср		
Базовые понятия и конструкции языка Си. Препроцессорные директивы.	8	Написание простейших программ на языке Си.	аудиторная работа, лабораторная работа
	8		
	12		
Функции.	6	Реализация пользовательских функций на языке Си.	аудиторная работа, лабораторная работа
	4		
	8		
Указатели, массивы, строки. Статическая и динамическая память.	12	Навыки работы со статической и динамической памятью на языке Си.	аудиторная работа, лабораторная работа
	16		
	22		
Организация ввода/вывода в языке Си. Файлы.	8	Навыки программного создания, чтения и изменения файлов на языке Си.	аудиторная работа, лабораторная работа
	8		
	12		
Стандартная библиотека языка Си.	4	Владение семейством функций стандартной библиотеки языка Си.	аудиторная работа, лабораторная работа
	4		
	6		
Структурные типы данных языка Си.	4	Определение пользовательских типов данных и операций над ними.	аудиторная работа, лабораторная работа
	8		
	8		
Абстрактные типы данных.	10	Реализация некоторых абстрактных типов данных на языке Си.	аудиторная работа, лабораторная работа
	12		
	16		
Алгоритмы сортировки и поиска данных.	6	Понимание основных методов сортировки данных и критериев их эффективности.	аудиторная работа, лабораторная работа
	8		
	10		
Парадигмы программирования. Общие принципы разработки программного обеспечения.	4	Знание основных парадигм программирования и общих принципов разработки ПО.	
	0		
	0		

² Не заполняется для ПУД, которые не вошли в УП ОП и не запланированы в расписании учебных занятий

Часов по видам учебных занятий:	62
	68
	98
Итого часов:	228

Формы учебных занятий:

лк – лекции в аудитории;

см - семинары/ практические занятия/ лабораторные работы в аудитории;

onl – лекции или иные виды работы студента с помощью онлайн-курса;

ср – самостоятельная работа студента.

3. Оценивание

Оценки по всем формам текущего контроля выставляются по 10-ти балльной шкале.

Для формирования практических навыков программирования проводятся лабораторные работы, семинары и выполняются домашние задания. При этом студент должен продемонстрировать не только уровень знаний, но и результаты самостоятельной работы: стремление к выполнению профессиональной деятельности, способность к поиску информации, использование для выполнения заданий знаний законов естественнонаучных дисциплин и компьютерных технологий.

Основной целью проведения лабораторных работ и семинаров является знакомство студентов с архитектурой современных ЭВМ, а также приобретение опыта практической разработки программ на языке Си. При выполнении лабораторных работ закрепляются навыки проектирования, программирования и документирования решений прикладных, в том числе математических, задач с помощью ЭВМ.

С целью текущего контроля успеваемости предусмотрена защита лабораторных работ.

- Оценка за лабораторную работу учитывает:
- Насколько точно студент выполнил задание, сформулированное в лабораторной работе;
- Степень и полноту усвоенных навыков работы со средствами разработки и отладки программ;
- Насколько слушатель правильно и аргументировано ответил на все вопросы при обсуждении выполненного задания;

- Качество оформления документации к программе и/или отчета по решению поставленной задачи.

Промежуточная оценка за 1 модуль – среднее арифметическое оценок за все выполненные лабораторные работы (N штук), аудиторную работу и домашнее задание:

$$O1 = (O_{\text{ауд}} + O_{\text{дз}} + \sum_{k=1}^n O_{\text{лр}i}) / (n + 2)$$

Промежуточная оценка за 2 модуль учитывает все выполненные лабораторные работы, контрольную и аудиторную работы второго модуля, O1 и оценку за экзамен:

$$O2 = 0.4 * (0.5 * O1 + 0.5 * (O_{\text{кр}} + O_{\text{дз}} + \sum_{k=1}^n O_{\text{лр}i}) / (n + 2)) + 0.6 * O_{\text{ЭКЗ}1}$$

Промежуточная оценка за 3 модуль – среднее арифметическое оценок за все выполненные лабораторные работы, аудиторную работу и домашнее задание 3-го модуля:

$$O3 = (O_{\text{ауд}} + O_{\text{дз}} + \sum_{k=1}^n O_{\text{лр}i}) / (n + 2)$$

Промежуточная оценка за 4 модуль учитывает все выполненные лабораторные работы, контрольную и аудиторную работы четвертого модуля и O2, O3:

$$O4 = 0.5 * [(O_{\text{кр}} + O_{\text{дз}} + \sum_{k=1}^n O_{\text{лр}i}) / (n + 2) + O3] + 0.5 * O2$$

Итоговая оценка учитывает оценку O4 и оценку за итоговый экзамен.

$$O_{\text{итог}} = 0.4 * O4 + 0.6 * O_{\text{ЭКЗ}2}.$$

Блокирующие элементы контроля отсутствуют.

4. Примеры оценочных средств

1. Парадигмы и языки программирования
2. Понятие алгоритма. Свойства алгоритма. Основные алгоритмические структуры. Способы представления алгоритма.
3. Структура программы на Си. Реализация принципа декомпозиции и модульности программ в языке Си.
4. Концепция типов для данных в рамках структурного программирования. Классификация данных.
5. Стандартные скалярные типы данных, операции над ними и их представление в памяти ЭВМ.
6. Массивы. Описание в программе и представление в памяти ЭВМ. Алгоритмы обработки и модификации массивов.
7. Представление строк в памяти ЭВМ. Библиотеки стандартных функций обработки символьных массивов.

8. Функции. Отличие между определением и описанием. Формальные и фактические параметры. Указатели на функции.
9. Рекурсия. Рекурсивные функции. Рекуррентные соотношения.
10. Препроцессорные средства.
11. Файлы. Сходство и различие массива и файла. Классификация файлов по типу и по способу доступа.
12. Основные приемы и функции работы с бинарными файлами.
13. Текстовые файлы и их особенности.
14. Стеки и очереди. Способы их представления на Си.
15. Связные списки и их структура.
16. Создать связный список, начиная с первого элемента списка.
17. Создать связный список, начиная с последнего элемента списка.
18. Печать элементов связанного списка (односвязного и двусвязного).
19. Добавить элемент в список (односвязный и двусвязный) после заданного значения.
20. Удалить элемент из списка (односвязного и двусвязного) по заданному значению.
21. Рекурсия по данным. Бинарное дерево. Способ представления его на Си. Создание упорядоченного бинарного дерева.
22. Печать вершин бинарного дерева в глубину с использованием стека.
23. Печать вершин бинарного дерева в ширину с использованием очереди.
24. Рекурсивный обход бинарных деревьев.
25. Поиск значения в связном списке.
26. Последовательный и бинарный поиск в массиве.
27. Связь между именами, адресами и значениями переменных. Указатели, адресная арифметика Динамическая память, библиотечные функции для работы с ней: распределения и освобождение памяти.
28. Массивы и указатели. Использование указателей при обработке массивов. Динамические массивы.
29. Локальные и глобальные переменные программы. Способы обмена данными между программой и функциями.
30. Графы. Основные определения. Способы представления графов. Алгоритмы обработки графов.

5. Ресурсы

5.1. Рекомендуемая основная литература

№п/п	
	1. Подбельский В. В., Фомин С. С. Курс программирования на языке Си. — М.: Финансы и статистика, 2018 (2012).
	2. Подбельский В. В. Практикум по программированию на языке Си. — М.: Финансы и статистика, 2004.
	3. Брайан Керниган, Деннис Ритчи Язык программирования С. — М.: Вильямс, 2017 (2015).

	<p>4. Вирт Н. Алгоритмы и структуры данных. ДМК Трест.2013.</p> <p>5. Стивен Прата Язык программирования С. Лекции и упражнения. – М.: Вильямс, 2018.</p>
--	---

5.2. Рекомендуемая дополнительная литература

№п/п	<p>1. Дейтел Х. М., Дейтел П. Дж. Как программировать на С. — М.: Бином-Пресс,2009.</p> <p>2. Демидович Е. М. Основы алгоритмизации и программирования. Язык Си. — СПб.: БХВ-Петербург,2008.</p> <p>3. Р. Седжвик. Фундаментальные алгоритмы на Си. — СПб: ООО «ДиаСофтЮП», 2003.</p> <p>4. Самюел П. Харбисон, Гай Л. Стил Язык С с примерами. — М.: Бином,2011.</p>
------	---

5.3. Программное обеспечение

№п/п	Наименование	Условия доступа/скачивания
	<p>Рекомендуемые компиляторы языка Си:</p> <ul style="list-style-type: none"> • GNU Compiler Collection версии не ниже 4.4 • Clang версии не ниже 3.1 • Microsoft C/C++ Optimizing Compiler версии не ниже 12 • Intel C++ Compiler версии не ниже 12 	<p><i>свободное лицензионное соглашение</i></p>

5.4. Профессиональные базы данных, информационные справочные системы, интернет-ресурсы (электронные образовательные ресурсы)
Не требуются.

5.5. Материально-техническое обеспечение дисциплины

Для успешного освоения дисциплины необходимо следующее материально-техническое обеспечение:

- Дисплейный класс, оборудованный современными персональными компьютерами

- Интерактивная доска и/или проектор с экраном.

6. Особенности организации обучения для лиц с ограниченными возможностями здоровья и инвалидов

В случае необходимости, обучающимся из числа лиц с ограниченными возможностями здоровья (по заявлению обучающегося) а для инвалидов также в соответствии с индивидуальной программой реабилитации инвалида, могут предлагаться следующие варианты восприятия учебной информации с учетом их индивидуальных психофизических особенностей, в том числе с применением электронного обучения и дистанционных технологий:

6.1.1. *для лиц с нарушениями зрения:* в печатной форме увеличенным шрифтом; в форме электронного документа; в форме аудиофайла (перевод учебных материалов в аудиоформат); в печатной форме на языке Брайля; индивидуальные консультации с привлечением тифлосурдопереводчика; индивидуальные задания и консультации.

6.1.2. *для лиц с нарушениями слуха:* в печатной форме; в форме электронного документа; видеоматериалы с субтитрами; индивидуальные консультации с привлечением сурдопереводчика; индивидуальные задания и консультации.

6.1.3. *для лиц с нарушениями опорно-двигательного аппарата:* в печатной форме; в форме электронного документа; в форме аудиофайла; индивидуальные задания и консультации.