

Документ подписан простой электронной подписью
Информация о владельце:
ФИО: Ястребов Олег Александрович
Должность: Ректор
Дата подписания: 20.05.2024 16:08:11
Уникальный программный ключ:
ca953a0120d891083f939673078ef1a989dae18a

**Федеральное государственное автономное образовательное учреждение высшего образования
«Российский университет дружбы народов имени Патриса Лумумбы»**

Инженерная академия

(наименование основного учебного подразделения (ОУП)-разработчика ОП ВО)

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ

ТЕХНОЛОГИИ ПРОГРАММИРОВАНИЯ

(наименование дисциплины/модуля)

Рекомендована МССН для направления подготовки/специальности:

02.04.02 ФУНДАМЕНТАЛЬНАЯ ИНФОРМАТИКА И ИНФОРМАЦИОННЫЕ ТЕХНОЛОГИИ

(код и наименование направления подготовки/специальности)

Освоение дисциплины ведется в рамках реализации основной профессиональной образовательной программы высшего образования (ОП ВО):

АНАЛИЗ БОЛЬШИХ ДАННЫХ И ТЕХНОЛОГИИ ЗАЩИТЫ ИНФОРМАЦИИ

(наименование (профиль/специализация) ОП ВО)

2024 г.

1. ЦЕЛЬ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Дисциплина «Технологии программирования» входит в программу магистратуры «Анализ больших данных и технологии защиты информации» по направлению 02.04.02 «Фундаментальная информатика и информационные технологии» и изучается в 1 семестре 1 курса. Дисциплину реализует Кафедра механики и процессов управления. Дисциплина состоит из 26 разделов и 65 тем и направлена на изучение базовых алгоритмов сортировки и поиска, алгоритмов на графах, методов динамического программирования, современных парадигм программирования, подходами к технологиям параллельного и распределенного программирования

Целью освоения дисциплины является овладение студентами практическими навыками алгоритмизации и программирования.

2. ТРЕБОВАНИЯ К РЕЗУЛЬТАТАМ ОСВОЕНИЯ ДИСЦИПЛИНЫ

Освоение дисциплины «Технологии программирования» направлено на формирование у обучающихся следующих компетенций (части компетенций):

Таблица 2.1. Перечень компетенций, формируемых у обучающихся при освоении дисциплины (результаты освоения дисциплины)

Шифр	Компетенция	Индикаторы достижения компетенции (в рамках данной дисциплины)
ОПК-2	Способен применять компьютерные/суперкомпьютерные методы, современное программное обеспечение (в том числе отечественного производства) для решения задач профессиональной деятельности	ОПК-2.1 Знает основные концепции современных вычислительных систем и программного обеспечения (в том числе отечественного производства); ОПК-2.2 Использует методы высокопроизводительных вычислительных технологий, современного программного обеспечения для решения задач профессиональной деятельности; ОПК-2.3 Использует инструментальные средства высокопроизводительных вычислений в научной и практической деятельности.;
ОПК-5	Способен устанавливать и сопровождать программное обеспечение информационных систем, осуществлять эффективное управление разработкой программных средств и проектов	ОПК-5.1 Знает порядок и особенности процесса инсталляции программного обеспечения информационных систем;; ОПК-5.2 Умеет обеспечить сопровождение программного обеспечения информационных систем;; ОПК-5.3 Владеет современными информационными технологиями и техническими средствами для осуществления эффективного управления разработкой программных средств и проектов.;

3. МЕСТО ДИСЦИПЛИНЫ В СТРУКТУРЕ ОП ВО

Дисциплина «Технологии программирования» относится к обязательной части блока 1 «Дисциплины (модули)» образовательной программы высшего образования.

В рамках образовательной программы высшего образования обучающиеся также осваивают другие дисциплины и/или практики, способствующие достижению запланированных результатов освоения дисциплины «Технологии программирования».

Таблица 3.1. Перечень компонентов ОП ВО, способствующих достижению запланированных результатов освоения дисциплины

Шифр	Наименование компетенции	Предшествующие дисциплины/модули, практики*	Последующие дисциплины/модули, практики*
ОПК-2	Способен применять компьютерные/суперкомпьютерные методы, современное программное обеспечение (в том числе отечественного производства) для решения задач профессиональной деятельности		Научно-исследовательская работа (получение первичных навыков научно-исследовательской работы); Научно-исследовательская работа; Преддипломная практика; Защищенное программное обеспечение;
ОПК-5	Способен устанавливать и сопровождать программное обеспечение информационных систем, осуществлять эффективное управление разработкой программных средств и проектов		Научно-исследовательская работа (получение первичных навыков научно-исследовательской работы); Научно-исследовательская работа; Преддипломная практика; Технологические угрозы и системы обеспечения кибербезопасности; Geoinformation Systems and Applications; Интеллектуальные информационные системы;

* - заполняется в соответствии с матрицей компетенций и СУП ОП ВО

** - элективные дисциплины /практики

4. ОБЪЕМ ДИСЦИПЛИНЫ И ВИДЫ УЧЕБНОЙ РАБОТЫ

Общая трудоемкость дисциплины «Технологии программирования» составляет «5» зачетных единиц.

Таблица 4.1. Виды учебной работы по периодам освоения образовательной программы высшего образования для очной формы обучения.

Вид учебной работы	ВСЕГО, ак.ч.		Семестр(-ы)
			1
<i>Контактная работа, ак.ч.</i>	34		34
Лекции (ЛК)	17		17
Лабораторные работы (ЛР)	17		17
Практические/семинарские занятия (СЗ)	0		0
<i>Самостоятельная работа обучающихся, ак.ч.</i>	119		119
<i>Контроль (экзамен/зачет с оценкой), ак.ч.</i>	27		27
Общая трудоемкость дисциплины	ак.ч.	180	180
	зач.ед.	5	5

5. СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Таблица 5.1. Содержание дисциплины (модуля) по видам учебной работы

Номер раздела	Наименование раздела дисциплины	Содержание раздела (темы)		Вид учебной работы*
Раздел 1	Основные элементы синтаксиса языка Python	1.1	Базовый синтаксис языка Python 3. Модель памяти и основные типы данных	ЛК, ЛР
		1.2	Циклы и списки. Функции.	ЛК, ЛР
Раздел 2	Элементы теории алгоритмов	2.1	Понятие алгоритма. Машина Тьюринга. Вычислимость. Теория сложности. Возведение в степень: анализ алгоритма (умное возведение в степень).	ЛК, ЛР
		2.2	Задача о рюкзаке. Жадный алгоритм. Метод градиентного спуска как пример жадного алгоритма. Стратегия «Разделяй и властвуй». Рекурсивный алгоритм.	ЛК, ЛР
Раздел 3	Парадигмы программирования. Объектноориентированное программирование	3.1	Основные принципы программирования. Процедурное программирование.	ЛК, ЛР
		3.2	Объектно-ориентированное программирование (ООП). Функциональное программирование.	ЛК, ЛР
		3.3	Особенности ООП. Классы и объекты. Наследование. Реализация ООП в языке Python	ЛК, ЛР
Раздел 4	Алгоритмы сортировки и поиска	4.1	Сортировка выбором. Сортировка вставками. Сортировка «Методом Пузырька». Сортировка слиянием. Быстрая сортировка	ЛК, ЛР
		4.2	Нахождение медианы. Последовательный поиск. Методы сужения области. Сортировка в Python.	ЛК, ЛР
Раздел 5	Алгоритмы на графах	5.1	Графы и их анализ. Представление графов. Обход графа в глубину и ширину	ЛК, ЛР
		5.2	Восстановление кратчайшего пути. Задача о перемещении шахматного коня	ЛК, ЛР
		5.3	Алгоритм Дейкстры. Очередь и стек. Очередь и стек в Python	ЛК, ЛР
Раздел 6	Динамическое программирование	6.1	Принцип оптимальности Беллмана. Понятие восходящего и нисходящего решения.	ЛК, ЛР
		6.2	Задача о количестве маршрутов. Сходства и отличие динамического программирования и концепция «разделяй и властвуй»	ЛК, ЛР
		6.3	Задача о банкомате. Динамическое программирование и игры	ЛК, ЛР
Раздел 7	Параллельные алгоритмы	7.1	Предпосылки. Классификация вычислительных систем. CPU и GPU процессоры.	ЛК, ЛР
		7.2	Характеристики параллельных алгоритмов. Типы непоследовательного программирования в Python.	ЛК, ЛР
		7.3	Процессы и Поток в Python. Асинхронные программы.	ЛК, ЛР
Раздел 8	Оптимизация программ	8.1	Методы оптимизации и ускорения программ на Python. Профилирование программ на языке Python.	ЛК, ЛР
		8.2	Модуль line_profiler. Компиляция Python: Ahead-of-time и Just-in-time компиляция. Модуль Numba.	ЛК, ЛР
		8.3	Cython как расширение языка Python. Особенности разработки программы на Cython	ЛК, ЛР
Раздел 9	C\C++. Введение	9.1	C и C++ особенности языков, история и эволюция. Машинноориентированные языки программирования и принципы действия компьютера.	ЛК, ЛР

Номер раздела	Наименование раздела дисциплины	Содержание раздела (темы)		Вид учебной работы*
		9.2	Трансляция кода. Виды трансляции. Отличия интерпретаторов и компиляторов.	ЛК, ЛР
		9.3	Сопоставление программ на Python и C/C++. Область применения и языков C/C++.	ЛК, ЛР
Раздел 10	Основные элементы синтаксиса	10.1	Блочное устройство программ на языках C/C++, синтаксические правила выделения блоков и их типы.	ЛК, ЛР
		10.2	Базовые инструкции: ветвление (или условная инструкция), циклы (while, do while и for), оператор безусловного перехода, оператор множественного выбора.	ЛК, ЛР
		10.3	Синтаксические конструкции для работы с функциями: объявление, определение, вызов. Стек вызовов. Сравнение goto и return	ЛК, ЛР
Раздел 11	Массивы и указатели	11.1	Указатели и адреса. Работа с указателями и адресами. Массив как структура данных: хранение в памяти, доступ к элементам.	ЛК, ЛР
		11.2	Создание статических массивов. Адресная арифметика	ЛК, ЛР
Раздел 12	Статическая и динамическая память.	12.1	Правила создание статических массивов, его инициализация и использование. Одномерные и многомерные статические массивы. Динамическая память (C стиль).	ЛК, ЛР
		12.2	Динамическая память (C++ стиль). Функции для работы с динамической памятью, операции выделения и освобождения памяти	ЛК, ЛР
		12.3	Создание одномерных и многомерных динамических массивов	ЛК, ЛР
Раздел 13	Структурированные типы данных	13.1	Массивы, строки символов, структуры, объединение, перечислимый тип данных, битовые поля. Синтаксические особенности объявления, инициализации и работы.	ЛК, ЛР
		13.2	Особенности «упаковки» в памяти. Примеры использования. Динамические структуры данных: вектор, очередь (стек), список, как примеры организации работы с структурированными данными в динамическом режиме.	ЛК, ЛР
Раздел 14	Перехват ошибок	14.1	Синтаксис операции обработки исключений. Примеры использования.	ЛК, ЛР
Раздел 15	Ввод-вывод данных	15.1	Понятие потока и буфера. Клавиатура, экран и файл как источник и приёмник данных. Организация потоков ввода и вывода данных в C++.	ЛК, ЛР
		15.2	Запись данных в поток и чтение данных из потока. Позиционирование данных в потоке. Режимы работы с файлами: чтение-запись, символьный/текстовый формат и их комбинации	ЛК, ЛР
		15.3	Текстовые и бинарные файлы, и особенность в них хранения данных. Файлы прямого доступа	ЛК, ЛР
Раздел 16	Объектно-ориентированное программирование в C++	16.1	Создание классов и объектов. Настройка модификаторов доступа: public, private и protected. Дружественные функции и классы.	ЛК, ЛР
		16.2	Ключевое слово this. Организация операции наследования в языке C++. Виртуальные функции и перегрузка функций и операторов.	ЛК, ЛР
Раздел 17	Использование библиотек	17.1	Обзор и примеры использования STL и BOOST	ЛК, ЛР
Раздел	Параллельные алгоритмы	18.1	Классификация вычислительных систем. CPU и	ЛК, ЛР

Номер раздела	Наименование раздела дисциплины	Содержание раздела (темы)		Вид учебной работы*
18	и системы		GPU процессоры. Характеристики параллельных алгоритмов	
		18.2	Типы непоследовательного программирования. Стандарты параллельных вычислений: взаимодействие между узлами суперкомпьютера, взаимодействие между ядрами одного CPU внутри одного узла, ускорители внутри одного узла	ЛК, ЛР
Раздел 19	Алгоритмы во внешней памяти	19.1	Организация вычислений с учётом иерархической структуры памяти. Буферизация при чтении и записи.	ЛК, ЛР
		19.2	Сложные и динамические структуры данных. Алгоритмы на графах во внешней памяти (BFS, DFS, поиск связанных компонент, MST)	ЛК, ЛР
Раздел 20	Технология OpenMP	20.1	Параллельные вычисления с использованием стандарта OpenMP.	ЛК, ЛР
		20.2	Основные сведения. Нити и процессы. Параллельные и последовательные области	ЛК, ЛР
		20.3	Параллельные циклы и параллельные области. Автоматическое распараллеливания циклов.	ЛК, ЛР
Раздел 21	Технология MPI	21.1	Параллельные вычисления с использованием стандарта MPI. Основные сведения. Основные процедуры MPI.	ЛК, ЛР
		21.2	Типы данных MPI. Способы передачи сообщений. Прием и передача сообщений процессами	ЛК, ЛР
Раздел 22	Технология OpenACC	22.1	Параллельные вычисления с использованием стандарта OpenACC	ЛК, ЛР
		22.2	Обзор производительности GPU в различных приложениях. Сравнение вычислительных ускорителей. Основные принципы достижения высокой производительности	ЛК, ЛР
		22.3	Преимущества OpenACC. Модель исполнения: gangs, workers, vectors. Директивы parallel, kernels, loop	ЛК, ЛР
		22.4	Атрибуты данных. Регионы данных: data, enter data, exit data. Дополнительные конструкции управления данными: cache, update, declare	ЛК, ЛР
		22.5	Асинхронное исполнение - async и wait. Атомарные операции. Глобальные переменные. OpenACC в C++	ЛК, ЛР
Раздел 23	Программно-аппаратная архитектура CUDA	23.1	Архитектура GPU. Иерархия памяти GPU. Программная модель CUDA.	ЛК, ЛР
		23.2	Использование библиотек C++ для программирования на OpenCL и CUDA	ЛК, ЛР
Раздел 24	Введение в распределенные объектные технологии	24.1	Понятие распределенной системы обработки информации. Виды и свойства распределенных систем.	ЛК, ЛР
		24.2	Архитектура программного обеспечения информационных систем. Управление взаимодействием разнородных приложений	ЛК, ЛР
		24.3	Основные механизмы распределенных объектных технологий.	ЛК, ЛР
Раздел 25	Основные модели распределенных объектных технологий	25.1	Виды распределенных приложений. Облачные технологии. Определение облачных вычислений. Многослойная архитектура облачных приложений. Компоненты облачных приложений.	ЛК, ЛР
		25.2	Достоинства и недостатки облачных	ЛК, ЛР

Номер раздела	Наименование раздела дисциплины	Содержание раздела (темы)		Вид учебной работы*
			вычислений. Наиболее распространенные облачные платформы. GRID-технологии.	
		25.3	Архитектура GRID. Стандарты GRID. Параметрические модели производительности GRID. Сравнение GRID и Облачных вычислений.	ЛК, ЛР
Раздел 26	Проблемы интеграции приложений	26.1	Проблемы интеграции приложений. Комплексная интеграция приложений. Брокеры сообщений. Модель взаимодействия "публикация/подписка"	ЛК, ЛР
		26.2	Системы управления рабочим потоком. Серверы приложений.	ЛК, ЛР

* - заполняется только по **ОЧНОЙ** форме обучения: ЛК – лекции; ЛР – лабораторные работы; СЗ – практические/семинарские занятия.

6. МАТЕРИАЛЬНО-ТЕХНИЧЕСКОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Таблица 6.1. Материально-техническое обеспечение дисциплины

Тип аудитории	Оснащение аудитории	Специализированное учебное/лабораторное оборудование, ПО и материалы для освоения дисциплины (при необходимости)
Лекционная		
Компьютерный класс		
Для самостоятельной работы		

* - аудитория для самостоятельной работы обучающихся указывается **ОБЯЗАТЕЛЬНО!**

7. УЧЕБНО-МЕТОДИЧЕСКОЕ И ИНФОРМАЦИОННОЕ ОБЕСПЕЧЕНИЕ ДИСЦИПЛИНЫ

Основная литература:

1. Python 3. Самое необходимое. Прохоренок Н., Дронов В., БХВ-Петербург, 2019 – 610 с.;
2. Python. Экспресс-курс. Седер Н., СПб.: Питер, 2019 – 480 с.;
3. Алгоритмы. Справочник с примерами на C, C++, Java и Python. Хайнеман Дж., Поллис Г., Селков С., СПб.: ООО "Альфа-книга", 2017 – 432 с.;
4. Программирование на языке высокого уровня. C/C++. Хабибуллин И.Ш., СПб.: БХВПетербург, 2006 – 512 с
5. C-C++. Справочник программиста. Г. Шилдт, Вильямс, 2003 - 429 с.;
6. Программирование на C++ в Visual Studio 2010 Express. Прохоренок Н.А., 2010 – 71 с
7. Язык программирования C. Брайан У. Керниган, Д.М. Ритчи, Вильямс, 2015 – 288 с
8. Язык программирования C++. Страуструп Б., Мартынов Н.Н., Москва: Бином, 2011. - 1135 с.
9. Параллельные вычисления. Воеводин В. В., Воеводин Вл. В., СПб.: БХВ-Петербург, 2002

10. Параллельное и распределенное программирование с использованием C++. Хьюз К., Хьюз Т., М.: Издательский дом «Вильямс», 2004;
11. Параллельное программирование с использованием OpenMP. Левин М.П. М.: Бином. Лаборатория знаний, 2008
12. Parallel programming with OpenACC. Farber R., Newnes, 2016 – 316 с.;
13. Технология CUDA в примерах: введение в программирование графических процессоров. Сандере Дж., Кэндрот Э. М.: ДМК Пресс, 2011 - 232 с
14. Распределенные системы. Принципы и парадигмы. Таненбаум Э., ван Стеен М. СПб: Питер, 2003. - 877 с
15. Разработка распределенных приложений на платформе .NET Framework. М. Сара, Р. Билл, Х. Шеннон, Б. Марк. СПб: Питер, 2008. - 608 с

Дополнительная литература:

1. Автоматизация рутинных задач с помощью Python: практическое руководство для начинающих. Свейгарт Эл., М.: "ИД Вильямс", 2017 – 592 с
2. Численные методы: Вычислительный практикум. Вабищевич П.Н., М.: «ЛИБРОКОМ», 2010 – 320 с.;
3. Язык программирования С. Лекции и упражнения. С. Прата, М.: Издательский дом «Вильямс», 2013 – 960 с.;
4. C++. Священные знания. Дьюхерст С., СПб.: Символ Плюс, 2012 – 240 с
5. Алгоритмы. Справочник с примерами на C, C++, Java и Python. Хайнеман Дж., Поллис Г., Селков С., СПб.: ООО "Альфа-книга", 2017 – 432 с.
6. Алгоритмы построение, анализ и реализация на языке программирования Си. Ворожцов А.В., Винокуров Н.А., Москва: МФТИ, 2007 – 452 с.
7. Программирование и информатика. Антонюк В.А., Иванов А.П., Москва: Физический фак. МГУ им. М. В. Ломоносова, 2015 – 64 с.
8. Последовательные и параллельные алгоритмы. Миллер Р., Боксер Л. М.: Бином. Лаборатория знаний, 2006
9. Введение в параллельные методы решения задач. Якобовский М. В. М.: Издательство Московского университета, 2013 – 328 с.;
10. Отладка приложений для Microsoft .NET и Microsoft Windows. Р. Джон. М.: Microsoft Press. Русская Редакция. 2008.
11. XML. Новые перспективы WWW. Бумфрей Ф, Диренцо О, Дакетт Й. Издательство: "ДМК Пресс", 2006. - 688 с
12. Чертовской В. Д. Базы данных: теория и практика: учебник для бакалавров. Советов Б. Я., Цехановский В. В. М.: ЮРАЙТ, 2011. - 459 с

Ресурсы информационно-телекоммуникационной сети «Интернет»:

1. ЭБС РУДН и сторонние ЭБС, к которым студенты университета имеют доступ на основании заключенных договоров
 - Электронно-библиотечная система РУДН – ЭБС РУДН
<http://lib.rudn.ru/MegaPro/Web>
 - ЭБС «Университетская библиотека онлайн» <http://www.biblioclub.ru>
 - ЭБС Юрайт <http://www.biblio-online.ru>
 - ЭБС «Консультант студента» www.studentlibrary.ru
 - ЭБС «Троицкий мост»
2. Базы данных и поисковые системы
 - электронный фонд правовой и нормативно-технической документации
<http://docs.cntd.ru/>
 - поисковая система Яндекс <https://www.yandex.ru/>
 - поисковая система Google <https://www.google.ru/>
 - реферативная база данных SCOPUS
<http://www.elsevier.com/locate/scopus/>

<http://www.elsevier.com/locate/scopus/>

Учебно-методические материалы для самостоятельной работы обучающихся при

освоении дисциплины/модуля:*

1. Курс лекций по дисциплине «Технологии программирования».

* - все учебно-методические материалы для самостоятельной работы обучающихся размещаются в соответствии с действующим порядком на странице дисциплины **в ТУИС!**

8. ОЦЕНОЧНЫЕ МАТЕРИАЛЫ И БАЛЛЬНО-РЕЙТИНГОВАЯ СИСТЕМА ОЦЕНИВАНИЯ УРОВНЯ СФОРМИРОВАННОСТИ КОМПЕТЕНЦИЙ ПО ДИСЦИПЛИНЕ

Оценочные материалы и балльно-рейтинговая система* оценивания уровня сформированности компетенций (части компетенций) по итогам освоения дисциплины «Технологии программирования» представлены в Приложении к настоящей Рабочей программе дисциплины.

* - ОМ и БРС формируются на основании требований соответствующего локального нормативного акта РУДН.

РАЗРАБОТЧИК:

Доцент

Должность, БУП

Подпись

Каратунов Максим

Олегович

Фамилия И.О.

РУКОВОДИТЕЛЬ БУП:

Заведующий кафедрой

Должность БУП

Подпись

Разумный Юрий

Николаевич

Фамилия И.О.

РУКОВОДИТЕЛЬ ОП ВО:

Доцент

Должность, БУП

Подпись

Варфоломеев Александр

Алексеевич

Фамилия И.О.