

## Программа учебной дисциплины «Алгоритмизация и программирование»

Утверждена

Академическим советом ОП

Протокол № 3 от 29.08.2019 г.

Автор	Ерохина Е.А., <a href="mailto:eerokhina@hse.ru">eerokhina@hse.ru</a>
Число кредитов	12
Контактная работа (час.)	140
Самостоятельная работа (час.)	316
Курс, образовательная программа.	1 курс, «Информатика и вычислительная техника».
Формат изучения дисциплины	с использованием онлайн-курса «Алгоритмизация вычислений» ( <a href="https://www.coursera.org/learn/algoritmizacija-vychislenij">https://www.coursera.org/learn/algoritmizacija-vychislenij</a> )

### 1. Цель, результаты освоения дисциплины и пререквизиты

Целями освоения дисциплины «Алгоритмизация и программирование» являются:

- освоение технологии структурного программирования;
- изучение базовых элементов языков Паскаль и C/C++;
- изучение стандартных алгоритмов обработки различных структур данных;
- изучение приёмов разработки алгоритмов для обработки различных структур данных.

В результате освоения дисциплины студент должен:

#### **знать:**

- основные типы алгоритмов;
- технологию структурного программирования;
- базовые элементы языков Pascal и C/C++;
- основы кодирования на языках программирования Pascal и C/C++ .

#### **уметь:**

- вводить исходные данные в программу и правильно интерпретировать результаты её работы;
- самостоятельно подбирать исходные данные для тестирования программ;
- использовать стандартное программное обеспечение в своей профессиональной деятельности.

#### **владеть навыками:**

- математической постановки задач;
- разработки алгоритмов решения задачи;
- трассировки алгоритмов;
- самостоятельного решения задач с использованием типовых алгоритмов и различных структур данных.

Изучение дисциплины «**Алгоритмизация и программирование**» базируется на следующих дисциплинах:

- математика в объеме средней школы;
- информатика и алгоритмизация в объеме средней школы.

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

- знать основы математики в пределах программы средней школы;
- знать основы информатики и алгоритмизации в пределах программы средней школы;
- уметь применять математический аппарат при выборе метода решения задачи.

Основные положения дисциплины должны быть использованы в дальнейшем при изучении следующих дисциплин.

1. Курсовая работа по дисциплине "Алгоритмизация и программирование".
2. Проектный семинар "Python в науке о данных".
3. Объектно-ориентированное программирование.
4. Теория автоматов и управление.
5. Базы данных.
6. Операционные системы.
7. Инструментальные средства и технология программирования.

## **2. Содержание учебной дисциплины**

В приведенной ниже таблице перечислены названия разделов дисциплины.

№	Название раздела
1.	Введение. Основы алгоритмизации.
2.	Основные алгоритмические конструкции. Язык Pascal.
3.	Циклы и примеры их использования
4.	Алгоритмы обработки одномерных массивов
5.	Алгоритмы обработки матриц
6.	Алгоритмы с досрочным выходом из цикла.
7.	Процедуры и функции.
8.	Символьный тип данных. Строковый тип данных.
9.	Файлы. Тип «множество».
10.	Динамические структуры в языке Pascal.
11.	Основные сведения о языке C.
12.	Конструкции языка C.
14.	Использование указателей при обработке массивов и матриц.
15.	Функции в языке C.
13.	Распределение памяти. Классы памяти.
16.	Символьный и строковый тип данных.
17.	Структуры. Файлы.
18.	Динамическое распределение памяти. Списки. Бинарные деревья.

Лекционный материал 1 и 2 модуля студенты частично изучают в рамках курса «Алгоритмизация вычислений» на платформе Coursera. Ниже приведены темы дистанционной части по неделям, разделенные на подтемы, а также указаны типы контроля.

Неделя. Тема недели	Номер темы в общем	Номер подтемы	Объединение по подтемам. Название подтемы	Номер клипа в лекции	Тема	Тип задания
1. Введение. Основы алгоритмизации	1.	1:01	Подходы к составлению алгоритмов и программ	1.1.	Преимущества и недостатки различных подходов к составлению алгоритмов и программ. Определение структурного программирования.	Тест

					Основные алгоритмические структуры (конструкции).
				1.2.	Общий вид постановки задачи. Метод решения задачи. Внешняя спецификация. Алгоритм. Программа.
				1.3.	Пример применения систематического подхода к составлению алгоритма и программы (решение линейного уравнения). Сравнение различных способов записи алгоритма (блок-схема, псевдокод, диаграмма Нейсси-Шнейдермана).
		1:02	Базовые операции и структуры (1.4-1.6)	1.4.	Общий вид алгоритма на псевдокоде. Понятие переменной. Классификация команд.
				1.5.	Базовые операции и базовые структуры. Операция присваивания.
				1.6.	Операция ввода. Операция вывода.

2. Основные алгоритмические конструкции. Язык Pascal.	2.	2:01	Алгоритмические конструкции	2.1.	Простая последовательность действий. Условная конструкция (выбор). Простейшая, расширенная условная конструкция, выбор. Вычисление номера четверти, которой принадлежит точка с заданными координатами x, y.	Тест
				2.2.	Циклы. От ... до (со счетчиком). Цикл-пока (с предусловием). Цикл-до (с постусловием). Взаимозаменяемость циклов «пока» и «до».	
		2:02	Основные сведения о языке Pascal	2.3.	Алфавит языка и особенности использования символов. Структура программы. Идентификаторы и служебные слова. Описание переменных.	
				2.4.	Классификация типов. Описание констант. Описание типов.	
		2:03	Операторы ввода и вывода. Стандартные процедуры и функции	2.5.	Операторы вывода. Форматный вывод. Ввод данных с клавиатуры.	
				2.6.	Стандартные процедуры и функции.	
		2:04	Операции в языке Pascal	2.7.	Арифметические и логические операции. Операции отношения. Простые и составные	

					операторы.	
				2.8.	Простейшая условная конструкция. Расширенная условная конструкция. Пример: программа для решения квадратного уравнения.	
3. Циклы и примеры их использования	3.	3:01	Циклы	3.1.	Цикл от... до (со счетчиком). Моделирование цикла с шагом отличным от 1 и -1 в языке Pascal.	Взаимное оценивание
				3.2.	Вычисление заданного количества чисел Фибоначчи.	
				3.3.	Кодирование циклов с предусловием и постусловием	
				3.4.	Пример использования цикла-до: метод половинного деления.	
		3:02	Итерационные алгоритмы	3.5.	Итерационные алгоритмы. Вывод рекуррентного отношения.	
				3.6.	Использование циклов с предусловием и постусловием при записи итерационных алгоритмов.	
		3:03	Одномерный массив	3.7.	Порядковые типы. Перечислимый тип. Интервальный тип. Описание массива. Ввод и вывод элементов одномерного массива	

		3:04	Алгоритмы вычисления минимума и максимума	3.8.	. Кванторы. Использование кванторов в постановке и методе решения задачи. Поиск максимального элемента и его номера в одномерном массиве из n элементов. Поиск максимума при помощи обращения по индексу. Поиск минимального по модулю элемента в одномерном массиве из n элементов.	
				3.9.	Поиск минимального среди элементов одномерного массива, удовлетворяющих заданному условию и его номера. Особенности поиска минимума (максимума) при дополнительном условии.	
4. Алгоритмы обработки одномерных массивов	4.	4:01	Сумма. Среднее арифметическое. Трассировка	4.1.	Вычисление суммы элементов массива, расположенных между первым и последним отрицательным элементом (включительно). Понятие трассировки. Поиск среднего арифметического положительных элементов массива.	Взаимное оценивание
		4:02	Формирование новых массивов	4.2.	Формирование нового массива с известным числом элементов. Формирование	

					нового массива с неизвестным числом элементов.	
				4.3.	Вычисление двух новых массивов неизвестной длины.	
		4:03	Алгоритмы удаления и вставки	4.4.	Удаление путем сдвига. Использование законов де Моргана для отрицания условий, содержащих неравенства и логические связки (или, и).	
				4.5.	Перестановка значений переменных. Алгоритмы удаления.	
				4.6.	Алгоритмы вставки.	
		4:04	Надежный ввод данных	4.7.	Обработка целых чисел. Надежный ввод данных. Разбиение числа на цифры с использованием функций div, mod.	
5. Алгоритмы сортировки. Обработка матриц.	4.	5:01	Алгоритмы сортировки	5.1.	Алгоритмы сортировки. Метод установки.	Взаимное оценивание
				5.2.	Сортировка методом пузырька. Сравнение методов.	
				5.3.	Поиск заданного количества максимальных элементов с использованием сортировки.	
	5.	5:02	Матрицы	5.4.	Матрицы. Ввод и вывод матрицы.	
				5.5.	Определение индексов элементов матрицы, расположенных на, над и под главной и побочной диагональю.	



				5.6.	Диагонали в прямоугольной матрице.	
				5.7.	Перемножение матриц.	
				5.8.	Использование методов сортировки при обработке матриц.	
6. Алгоритмы с досрочным выходом из цикла.	6.	6:01	Алгоритмы с досрочным выходом из цикла. Одномерный массив	6.1.	Алгоритмы с досрочным выходом из цикла. Использование логической переменной для организации досрочного выхода из цикла	Тест
				6.2.	Формирование массива, содержащего все повторяющиеся элементы исходного массива. Проверка того, что элемент не повторяется в массиве.	
				6.3.	Подсчет числа повторов каждого элемента массива	
				6.4.	Формирование массива, содержащего все различные элементы исходного массива по одному разу.	
	6:02	Алгоритмы с досрочным выходом из цикла. Матрица	6.5.	Поиск минимального среди элементов заданной целочисленной матрицы, отсутствующих в заданном одномерном массиве.		
			6.6.	Проверка присутствия для двух элементов массива.		

7. Процедуры и функции.	7.	7:01	Процедуры и функции	7.1.	Описание процедур и функций.	Тест
				7.2.	Раздел описаний процедуры или функции.	
	7:02	Параметры процедур и функций	7.3.	Параметры-переменные и параметры-значения.		
				Формальные и фактические параметры. Параметры-константы.		
			7.4.	Локальные и глобальные переменные. Побочные эффекты при изменении глобального параметра в теле подпрограммы.		
	7:03	Решение задач с использованием функций	7.5.	Подсчет числа размещений с использованием функции вычисления факториала.		
7.6.			Вычисление минимума из максимальных элементов строк матрицы с использованием функции.			
8. Решение задач с использованием процедур и функций.	7.	8:01	Рекурсия	8.1.	Рекурсия. Итерационный и рекурсивный алгоритм вычисления факториала. Трассировка рекурсивной функции. Вычисление чисел Фибоначчи с использованием рекурсии. Опережающее объявление.	Взаимное оценивание

		8:02	Процедуры и функции для одномерного массива	8.2.	Формирование нового массива, содержащего повторяющиеся элементы исходного массива с одним вхождением с использованием процедуры.	
				8.3.	Программа с использованием функции для вычисления нового массива, содержащего элементы тех строк матрицы, номера которых отсутствуют в заданном одномерном массиве.	
		8:03	Процедуры и функции для матрицы	8.4.	Программа с использованием подпрограммы для вычисления одномерного массива, содержащего заданное число наименьших элементов среди элементов квадратной целочисленной матрицы, расположенных ниже главной диагонали.	
				8.5.	Слияние упорядоченных массивов.	
		8:03	Нисходящее проектирование	8.6.	Пример нисходящего проектирования	
9. Символьный тип данных. Строковый тип	8	9:01	Стандартные процедуры и функции	9.1.	Стандартные процедуры для работы со строками.	Взаимное оценивание

данных. Записи		9:02	Решение задач обработки символьных строк	9.2.	Стандартные функции для работы со строками.		
					Решение задач обработки символьных строк. Правильность расстановки скобок в формуле.		
					Удаление парных пробелов и подсчет количества слов.		
					9.3.		Поиск символа с минимальным кодом.
					9.4.		Удваивание русских букв в заданной символьной строке.
					9.5.		Выделение из массива символьных строк подстрок, не содержащих цифр.
9:03	Записи	9.6.	Записи (тип record). Оператор with.				
10. Файлы. Тип «множество».	9.	10:01	Процедуры и функции для работы с файлом	10.1	Файлы. Стандартные процедуры для работы с файлами.	Взаимное оценивание	
					Стандартные функции для работы с файлами		
		10:02	Обработка текстовых и типизированных файлов	10.2	Поиск максимального отрицательного числа в типизированном файле.		
					10.3		Построчное чтение текстового файла и подсчёт общего числа символов и кол-ва цифр.
		10:03	Файл записей	10.4	Поиск самого молодого студента .		
					10.5		Пример обработки файла записей: поиск самого дорогого набора с

					заданным весом.	
				10.6	Упорядочение цифр, расположенных в каждой строке файла, по возрастанию.	
		10:04	Тип "множество"	10.7	Множества. Операции с множествами	
				10.8	Алгоритм «Решето Эратосфена».	
11. Статические и динамические переменные. Указатели. Связанные списки.	10	11:01	Указатели	11.1	Типизированный указатель. Операция разыменования.	Тест
				11.2	Стандартные процедуры для работы с памятью. Примеры использования указателей.	
		11:02	Связанные списки	11.3	Основные типы связанных списков стек и очередь, линейные и циклические связанные списки, -одно- и двунаправленные списки.	
				11.4	Формирование стека. Вывод списка на экран.	
				11.5	Формирование очереди.	
12. Статические и динамические переменные. Указатели. Связанные списки.	10	12:01	Создание нового списка	12.1	Формирование новой очереди из элементов списка кратных трём.	Взаимное оценивание
		12:02	Сортировка связанного	12.2	Сортировка связанного списка по	

			списка		убыванию методом пузырька.
				12.3	Сортировка связанного списка методом установки.
	12:03	Удаление и вставка элементов в связанном списке		12.4	Удаление элементов из списка.
				12.5	Вставка элементов в список.
	12:04	Реверс		12.6	Реверс списка.

В приведенной ниже таблице приведено распределение часов по темам курса.

Тема (раздел дисциплины)	объем в часах	Планируемые результаты обучения (ПРО), подлежащие контролю	Формы контроля
	лк		
	см		
	онл/ср		
Тема 1. Введение. Основы алгоритмизации.	3	Знает основные способы записи алгоритмов. Знает базовые конструкции (структуры) и базовые операции псевдокода.	Тест и вопросы в лекции в дистанционной части курса.
	12		
Тема 2. Основные алгоритмические конструкции. Язык Pascal.		Умеет записать постановку задачи. Знает классификацию базовых типов в языке Паскаль. Знает стандартные процедуры и функции. Арифметические и логические операции. Операции отношения. Простые и составные операторы.	Тест и вопросы в лекции в дистанционной части курса.
	14		

Тема 3. Циклы и примеры их использования		Знает типы циклов и умеет правильно записать их на псевдокоде. Может выбрать тип цикла, который подходит для решения данной задачи. Может составить итерационный алгоритм и закодировать его.	Вопросы в лекции в дистанционной части курса. Задание на взаимное оценивание в дистанционной части курса.
	16		
Тема 4. Алгоритмы обработки одномерных массивов	1	Умеет использовать алгоритмы сортировки, удаления, вставки элементов, поиска минимума или максимума, формирования нового массива для решения задач обработки одномерного массива.	Лабораторная работа 1 в очной части курса. Вопросы в лекции в дистанционной части курса. Задание на взаимное оценивание в дистанционной части курса.
	4		
	18		
Тема 5. Алгоритмы обработки матриц.		умеет осуществлять ввод матриц, а также использовать алгоритмы, перечисленные в п. 4 для обработки матриц.	Задание на взаимное оценивание в дистанционной части курса. Лабораторная работа 2 в очной части курса.
	20		
Тема 6. Алгоритмы с досрочным выходом из цикла.	3	Умеет использовать алгоритмы с досрочным выходом из цикла для обработки массивов и матриц.	Вопросы в лекции в дистанционной части курса. Тест в дистанционной части курса. Лабораторная работа 2 в очной части курса.
	12		
Тема 7. Процедуры и функции.	3	Знает синтаксис описания процедур и функций, а также механизм передачи параметров. Умеет выделять подпрограммы при решении задачи.	Вопросы в лекции в дистанционной части курса. Задание на взаимное оценивание в дистанционной
	5		

	20		части курса. Лабораторная работа 4 в очной части курса. Контрольная работа.
Тема 8. Символьный тип данных. Строковый тип данных. Записи	3	Знает основные сведения о символьном и строковом типе данных. Умеет использовать стандартные функции при обработке строк. Решает задачи на обработку символьных строк.	Вопросы в лекции в дистанционной части курса. Задание на взаимное оценивание в дистанционной части курса. Лабораторная работа 5 в очной части курса. Контрольная работа.
	5		
	18		
Тема 9. Файлы. Тип «множество».	3	Умеет использовать стандартные функции обработки файлов и работы с типом "множество". Умеет использовать стандартные алгоритмы для обработки этих типов данных.	Вопросы в лекции в дистанционной части курса. Задание на взаимное оценивание в дистанционной части курса. Лабораторная работа 5 в очной части курса.
	2		
	20		
Тема 10. Динамические структуры в языке Pascal.		Знает основные типы связанных списков. Умеет формировать и обрабатывать линейные однонаправленные списки - стек и очередь, т.е. применяет стандартные алгоритмы для работы с этим типом данных.	Вопросы в лекции и тест в дистанционной части курса. Задание на взаимное оценивание в дистанционной части курса.
	20		
Тема 11. Основные сведения о языке С.	2	Знает структуру простой программы на языке С. Знает основные типы данных, базовые операции (арифметические, логические и операции присваивания). Умеет кодировать	Проверочные работы на лекциях и семинарах.



	20	арифметические выражения на языке C.	
Тема 12. Конструкции языка C.	4	Знает кодирование всех алгоритмических конструкций на языке C.	Проверочные работы на лекциях и семинарах.
	2		
	16		
Тема 13. Использование указателей при обработке массивов и матриц.		Знает понятие и назначение типа "указатель". Умеет использовать указатели при решении задач обработки массивов и матриц.	Проверочные работы на лекциях и семинарах. Лабораторные работы 1 и 2.
	2		
	20		
Тема 14. Функции в языке C.	5	Знает понятие функции. Умеет передавать параметры по значению и по адресу. Умеет использовать указатель на указатель.	Проверочные работы на лекциях и семинарах. Лабораторные работы 1 и 2. Контрольная работа 1.
	10		
	20		
Тема 15. Распределение памяти. Классы памяти.	1	Знает характеристики классов памяти и основные отличия между ними.	Опрос на лекции.
	2		
Тема 16. Символьный и строковый тип данных.	3	Умеет использовать стандартные функции для обработки символьных строк при решении задач. Умеет использовать алгоритмы обработки смвольных сток при решении задач. Умеет использовать указатели для обработки символьных строк.	Проверочные работы на лекциях и семинарах. Лабораторная работа 3. Контрольная работа 2.
	6		
	22		

Тема 17. Структуры. Файлы.	5	Знает понятия структуры и файлового типа. Умеет применять стандартные алгоритмы для решения задач обработки файлов и структур.	Проверочные работы на лекциях и семинарах. Лабораторная работа 4. Контрольная работа 2.
	8		
	22		
Тема 18. Динамическое распределение памяти. Списки. Бинарные деревья.	8	Умеет использовать функции выделения и освобождения памяти. Умеет формировать и обрабатывать связанные списки различных типов - линейные, циклические, одно и двунаправленные. Умеет формировать и обрабатывать бинарные деревья.	Проверочные работы на лекциях и семинарах. Лабораторные работы 5 и 6. Контрольная работа 2.
	16		
	24		
<b>Часов по видам учебных занятий:</b>	<b>44</b>		
	<b>60</b>		
	<b>316</b>		
<b>Итого часов:</b>	<b>420</b>		

Далее приведены темы и краткое содержание лекций и семинаров для очной части обучения. Нумерация лекций, семинаров и лабораторных работ начинается заново для каждого полугодия, нумерация тем общая по всему курсу. Если номер темы отсутствует в списке, то весь материал этой темы изучается дистанционно.

#### Тема 1. Введение. Основы алгоритмизации.

##### Лекция 1.

Оценивание результатов обучения. Требования к оформлению отчета по лабораторным работам. Пример оформления отчета по лабораторной работе 1. Правила выбора заданий в соответствии с номером варианта. (2 ч.).

##### Лекция 2

Пример систематического подхода к разработке алгоритмов и программ (решение квадратного уравнения). Анализ программы на языке Паскаль. (1 ч.).

#### Тема 4. Алгоритмы обработки одномерных массивов

##### Лекция 2

Вычисление двух новых массивов неизвестной длины. Алгоритмы удаления элементов массива без использования дополнительного массива (путем сдвига). Алгоритм перестановки значений переменных. (1 ч.).

#### Семинар 1

Алгоритм поиска минимума (максимума) и их номеров. Использование кванторов при записи постановки задачи. Поиск минимума и максимума при наличии дополнительных условий. Формирование нового массива с известным числом элементов. Формирование нового массива с неизвестным числом элементов. Вычисление среднего арифметического элементов массива. (2 ч.).

#### Семинар 2

Алгоритм «Удаление путем сдвига». Алгоритмы вставки элементов массива в другой массив. Алгоритм и программа вычисления суммы элементов массива в указанном диапазоне. (2 ч.).

Тема 6. Алгоритмы с досрочным выходом из цикла.

#### Лекция 3

Поиск максимальной цифры заданного числа. Алгоритмы с досрочным выходом из цикла. Варианты организации досрочного выхода. Использование алгоритмов с досрочным выходом при обработке матриц. (2 ч.).

#### Лекция 4

Алгоритм удаления элементов массива с использованием досрочного выхода из цикла. (1 ч.).

## Тема 7. Процедуры и функции.

### Лекция 4

Процедуры и функции. Заголовок процедуры и функции. Формальные и фактические параметры. Принцип локализации переменных. Побочные эффекты при использовании глобальных переменных. Примеры программ с использованием подпрограмм для обработки массивов и матриц. (1 ч.).

### Лекция 5.

Рекурсивная функция для вычисления чисел Фибоначчи. Программа с использованием подпрограмм с использованием алгоритма с досрочным выходом из цикла для одномерного массива. Опережающее объявление подпрограмм. Пример разработки проекта с использованием нескольких подпрограмм. (2 ч.).

### Семинар 3

Процедуры и функции. Рекурсивный и итерационный алгоритм вычисления факториала (программа с использованием функции для вычисления числа размещений). Вычисление минимума из максимальных по модулю элементов столбцов матрицы с использованием функции. Алгоритм с досрочным выходом и программа с использованием процедур для проверки повторений элементов одномерного массива. Формирование нового массива, содержащего неповторяющиеся элементы исходного массива с одним вхождением с использованием процедуры. (2 ч.).

### Семинар 4

Формирование нового упорядоченного массива из элементов двух исходных упорядоченных массивов. Программа с использованием функции для вычисления нового массива, содержащего элементы тех строк матрицы, номера которых повторяются в заданном одномерном массиве. Программа с использованием подпрограммы для вычисления одномерного массива, содержащего заданное число наименьших элементов среди элементов квадратной целочисленной матрицы, расположенных выше побочной диагонали. (2 ч.).

### Семинар 5

Удаление повторяющихся элементов массива с использованием процедуры. (1 ч.).

## Тема 8. Символьный тип данных. Строковый тип данных.

### Лекция 6.

Символьный тип данных. Строковый тип данных. Стандартные процедуры и функции для работы со строками. Решение задач обработки символьных строк. (2 ч.).

### Лекция 7.

Пример решения задачи лабораторной работы по обработке символьных строк. (1 ч.).

## Семинар 5

Поиск строки с минимальным числом цифр. Замена цифры на ее код. (1 ч.).

## Семинар 6.

Контрольная работа. (2 ч.).

## Семинар 7.

Обработка символьных строк. Разбор примера выполнения лабораторной работы по обработке символьных строк. (2 ч.).

Тема 9. Файлы. Тип «множество».

## Лекция 7.

Записи (тип record). Оператор присоединения with. Файлы. Стандартные процедуры и функции для работы с файлами. Решение задач обработки файлов. (1 ч.).

## Лекция 8.

Файлы. Записи. Решение задач. (2 ч.).

## Семинар 8.

Файлы. Удаление отрицательных чисел из файла путем сдвига. Сортировка строк файла по возрастанию их длин. Поиск самых длинных различных фамилий студентов в массиве записей. (2 ч.).

Тема 11. Основные сведения о языке C.

## Лекция 9.

Язык программирования C. Достоинства языка. Структура языка программирования. Форматы функций printf() и scanf(). Структура простой программы на Си. Простые и составные типы. Типы данных. Пример описания простых переменных. Определение размеров памяти для хранения данных. Константы целые, вещественные, символьные и строковые. Специальные управляющие символьные константы. Описания с начальным присваиванием. Выражения. Операции в Си: унарные, бинарные и тернарные. Арифметические и логические операции, операции отношения. Операции присваивания. Инкремент и декремент, префиксная и постфиксная форма инкремента и декремента. Операции с присваиванием. Множественные присваивания. Операция «запятая». Операция приведения типов. Кодирование формул. Условная конструкция. (2 ч.).

Тема 12. Конструкции языка C.

## Лекция 10.

Пример программы: решение квадратного уравнения. Операция условия. Множественный выбор. Оператор Break. Правила выбора условных операторов. Циклы. Цикл-пока (while). Цикл for. Замена for на while. Отличия for от Паскаля. Вложенные циклы. Совершенные числа. Цикл с постусловием do...while. Управляющие операторы break, continue, go to. (2 ч.).

## Лекция 11.

Примеры реализации стандартных алгоритмов обработки массивов и матриц на языке C. Основные возможности форматного ввода и вывода в языке C. Поточковый ввод и вывод данных в языке C++. (2 ч.).

## Семинар 9.

Язык C. Особенности объявлений и некоторые операции. Указатели в языке C. Массивы и матрицы. Варианты ввода одномерного массива. Решение задач обработки массивов с использованием индексов. (2 ч.).

Тема 13. Использование указателей при обработке массивов и матриц.

## Семинар 10.

Перестановка элементов массива в заданном порядке с сохранением порядка следования отрицательных и неотрицательных элементов. Формирование массива C, состоящего из повторяющихся элементов исходного массива A, отсутствующих в заданном массиве B. Использование алгоритмов с досрочным выходом при решении задач обработки одномерных массивов (с применением указателей). (2 ч.).

Тема 14. Функции в языке C.

## Лекция 1.

Определение индексов элементов матрицы, расположенных на, над и под главной и побочной диагоналями. Использование указателей при обработке матриц. Сортировка строк и столбцов матрицы. Синтаксические особенности Си. Функции. Описание функции. Правила оформления функций. Вызов функции. (2 ч.).

## Лекция 2.

Оператор return. Функция обмена значения двух переменных. Передача параметров по адресу и по значению. Правила оформления функций. Особенности работы функций. Пример использования функции – бинарный поиск в упорядоченном массиве. (1 ч.).

## Лекция 3.

Массивы и указатели. Операции с указателями. Многомерные массивы. Доступ к элементам массива с помощью указателей. Использование указателей при обработке массивов и матриц. Макросы. Флаги форматирования в C++. Манипуляторы форматирования в C++. (2 ч.).

## Семинар 1.

Перестановка двух последних нечетных элементов одномерного целочисленного массива с использованием указателей. Перестановка двух наименьших элементов матрицы  $A[0:n-1,0:m-1]$  с использованием указателей. Реализация методов сортировки одномерного массива с использованием указателей. (2 ч.).

## Семинар 2.

Обработка матриц с использованием указателей. Решение задач. Обмен строк матрицы. (2 ч.).

### Семинар 3.

Слияние упорядоченных массивов с использованием указателя и функции. Удаление повторяющихся элементов массива, отсутствующих в другом массиве, с использованием указателя и функции. Вычисление максимального среди отрицательных элементов целочисленной матрицы, не встречающихся в одномерном массиве. с использованием указателя и функции. (2 ч.).

### Семинар 4.

Обмен местами второго и пятого четных элементов матрицы с использованием указателей и функции для поиска адресов этих элементов (указатель на указатель), Формирование массива указателей на минимальный четный элемент каждой строки матрицы с использованием функции. Решение задач для подготовки к контрольной работе. (2 ч.).

### Семинар 5.

Контрольная работа. (2 ч.).

Тема 15. Распределение памяти. Классы памяти.

#### Лекция 2.

Распределение памяти. Классы памяти. Свойства классов памяти. Автоматические переменные. Внешние переменные. Атрибут extern. Статические переменные. Внешние статические переменные. Регистровые переменные. (1 ч.).

Тема 16. Символьный и строковый тип данных.

#### Лекция 4.

Операции над типом char. Ввод и вывод данных типа char. Символьные строки. Примеры использования указателя при работе со строками. Инициализация. Использование указателя для создания строки. Массивы символьных строк. Операции с типом char. Ввод-вывод строк. Функции scanf() и printf(). Функции gets и puts. Ввод строк до ввода пустой строки. Функции getchar() и putchar(). Строковые функции. (2 ч.).

#### Лекция 5.

Проверка правильности ввода исходных данных. Функции классификации. Решение задач обработки символьных строк с использованием указателей. (1 ч.).

### Семинар 6.

Символьные строки. Ввод массива символьных строк до пустой строки. Удаление парных пробелов с использованием указателей. Примеры подпрограмм, реализующих стандартные функции. Удаление подстрок. Поиск подстрок с заданными свойствами. (2 ч.).

### Семинар 7.

Пример решения заданий лабораторной работы на обработку массива символьных строк. (2 ч.).

## Семинар 8.

Особенности работы с кодировками символов в Qt Creator. Формирование массива из различных слов заданной строки. Замена символов в подстроках. Удаление подстрок путем сдвига. (2 ч.).

## Тема 17. Структуры. Файлы.

### Лекция 5.

Структуры. Объявление структурного типа. Определение структурных переменных. Инициализация. Доступ к элементам структуры, составные имена. Массивы структур. Вложенные структуры. Определение переменной структурного типа и инициализация. (1 ч.).

### Лекция 6.

Указатели на структуры. Указатели и массивы структур. Операции над структурами. Присваивание. Сравнение структур. Доступ к элементу структуры при помощи указателя. Доступ с помощью составного имени. Передача информации о структурах функциям. Текстовые файлы. Открытие и закрытие файла. Поточные функции ввода и вывода для файлов. Функции ввода и вывода (fgetc(), fputc()). Форматный ввод/вывод (fscanf(), fprintf()). Ввод строки (fgets()). Вывод строки (fputs()). Определение конца файла (feof). (2 ч.).

### Лекция 7.

Текстовые файлы. Объект типа FILE. Открытие и закрытие файла. Вывод на экран содержимого текстового файла. Функции ввода и вывода из файла. Ввод чисел и строк из файла. Функция fseek - установка позиции в текстовом файле ввода-вывода. Функция ftell. Использование функции exit. Чтение матрицы из файла. Сортировка слов в файле в лексикографическом порядке. (2 ч.).

## Семинар 9.

Поиск самой дорогой книги в массиве структур с использованием указателей. Формирование нового массива структур. (2 ч.).

## Семинар 10.

Обработка массива структур с информацией о геометрических фигурах. Сортировка массива структур с использованием массива указателей на структуры. (2 ч.).

## Семинар 11.

Ввод данных для массива структур с проверкой их допустимости. Формирование нового массива структур и указателей на структуры. (2 ч.).

## Семинар 12.

Файлы. Особенности текстовых файлов в языке Си. Чтение массива из файла и поиск максимума. Чтение из файла и запись в файл массива структур. Варианты ввода массива структур. (2 ч.).



## Тема 18. Динамическое распределение памяти. Списки. Бинарные деревья.

### Лекция 8.

Функции динамического распределения памяти. Функция `malloc()`, `calloc()`, `free()`, `sizeof()`. Функции `new`, `delete`. Динамическое выделение памяти для одномерного массива. Динамическое выделение памяти для двумерного массива. Динамические структуры данных. Формирование стека и вычисление суммы элементов стека. Циклический список. Подпрограмма занесения значения  $X$  в очередь на базе циклического списка. Двухнаправленные связанные списки. Занесение значения  $X$  в стек на базе линейного двухнаправленного связанного списка. Удаление элементов из циклического однонаправленного списка. Освобождение памяти, занятой связанным списком. (2 ч.).

### Лекция 9.

Бинарные деревья. Алгоритм обхода бинарного дерева (в прямом, обратном, симметричном порядке). Формирование бинарного дерева. Использование бинарного дерева для сортировки. Свойства рекурсивных алгоритмов. Операции над бинарными деревьями. Вывод дерева на экран. Задачи обработки бинарного дерева: подсчет числа узлов, подсчет числа листьев, подсчет числа одностенных отцов, подсчет глубины. (2 ч.).

### Лекция 10.

Задачи обработки бинарного дерева: удаление листьев, поиск максимума, удаление чисел, меньших заданного числа. Использование бинарного дерева для вычисления значения арифметического выражения. Мультисписки. Формирование мультисписка, вывод списка на экран. (2 ч.).

### Лекция 11.

Решение задач для подготовки к сдаче экзамена. (2 ч.).

### Семинар 13.

Динамические структуры данных. Функции `malloc()` - выделение памяти, `free()` - освобождение памяти. Ввод текста в динамическую память. Освобождение памяти, занятой введенным текстом. Формирование стека, содержащего структуры КНИГА (исходные данные читаются из файла). (2 ч.).

### Семинар 14.

Формирование очереди. Сортировка однонаправленного списка. Удаление элементов без использования нового списка. (2 ч.).

### Семинар 15.

Вставка элементов в список. Реверс списка. (2 ч.).

### Семинар 16.

Контрольная работа. (2 ч.).

### Семинар 17.

Формирование двунаправленного связанного списка (очередь). Вставка элементов в двунаправленный список. Формирование нового списка из различных элементов исходного. (2 ч.).

### Семинар 18.

Бинарные деревья. Сортировка по возрастанию с использованием бинарного дерева. Удаление из данного бинарного дерева всех значений, равных заданному числу X. (2 ч.).

### Семинар 19.

Использование рекурсии для проверки правильности записи арифметического выражения. (2 ч.).

### Семинар 20.

Решение задач для подготовки к сдаче экзамена. (2 ч.).

## 3. Оценивание

### Формы контроля знаний студентов

Тип контроля	Форма контроля	Модуль				Параметры **
		1	2	3	4	
Текущий (неделя)	Контрольная работа		5	5	6	Письменная работа 80 минут
	Лабораторная работа	5,7	3,5,7	3,5,7	3,5,7	Решение задачи на компьютере с последующей защитой отчета. При несвоевременной защите отчета баллы за работу снижаются.
Промежуточный	Экзамен		*			Решение задачи на компьютере 90 минут. Результаты проверки в день сдачи экзамена
Итоговый	Экзамен				*	Решение задачи на компьютере 120 минут. Результаты проверки в день сдачи экзамена

Для вычисления текущей оценки по дисциплине (без учета результатов обучения на платформе Coursera) используется следующая таблица.

	Работа на семинарском занятии	Работа на лекции	Выполнение лабораторного практикума	Контрольная работа
1 модуль	2	1	7 (3+4)	-
2 модуль	1	1	5(2+2+1)	3
3 модуль	1	1	5(1,5+1,5+2)	3
4 модуль	1	1	5(2,3+1,7+1)	3

Ни один из элементов текущего контроля не является блокирующим.

Преподаватель оценивает работу студентов на лекциях и семинарских занятиях, выполнение лабораторных работ и своевременность защиты отчетов. Все эти виды работ влияют на оценку промежуточного контроля.

Преподаватель оценивает работу студентов на семинарских и практических занятиях следующим образом. На некоторых семинарских занятиях проводится оперативное оценивание усвоения текущего материала (тест или проверочная работа), либо проверка выполнения домашнего задания. Каждый вид работы оценивается от 1 до 6 баллов. В итоговую оценку эти баллы входят с коэффициентом, получаемым делением числа занятий, на которых проводилось оценивание, на общее количество занятий. При пропуске лекции или семинарского занятия по любой причине студент не может решить дополнительное задание для компенсации баллов, которые он мог получить на этом занятии.

Кроме того, преподаватель может оценивать дополнительными баллами ответ студента у доски и активное участие в решении задач семинаров (например, выявление и исправление неточностей и ошибок в алгоритмах и при кодировании программ, внесение усовершенствований в алгоритм и т.п.). При многочисленных ответах у доски и выполнении на семинарах заданий, оцениваемых дополнительно, текущая оценка успеваемости студента может превышать 10 баллов за модуль. Все эти дополнительные баллы учитываются в промежуточной оценке соответствующего модуля.

Оценки за работу на семинарских и практических занятиях преподаватель выставляет в журнал. Журнал хранится на яндекс-диске (ссылка приведена выше). Накопленная оценка за работу на семинарских занятиях определяется перед итоговым контролем.

Преподаватель оценивает самостоятельную работу студентов: выполнение лабораторных работ. Оценивается правильность выполнения работы и своевременность защиты отчета. Для каждой лабораторной работы устанавливается срок защиты отчета. При своевременной защите работа оценивается наивысшим баллом, при опоздании на 1 неделю балл снижается на

40%, при опоздании на 2 недели балл снижается на 60% от полученной оценки. При опоздании более чем на 2 недели работа не оценивается.

В случае пропуска занятий по уважительной причине (обязательно предоставление справки) срок сдачи соответствующей лабораторной работы может быть перенесен на соответствующее количество рабочих дней.

Вес каждой лабораторной работы в текущей оценке за модуль устанавливается отдельно для каждого модуля (соответствующие значения приведены в таблице, в графе «Выполнение лабораторного практикума»).

Оценки за самостоятельную работу студента преподаватель выставляет в журнал.

В случае пропуска занятий по уважительной причине (обязательно предоставление справки) предоставляется дополнительное время для написания контрольной работы (единственная дата переписывания заранее сообщается через старост).

Переписывание контрольной работы с целью повышения полученной оценки не допускается.

Оценка за текущий контроль в 1 и 2 модуле учитывает результаты студента по текущему контролю следующим образом.

1. Модуль №1.  $O_{\text{текущая } 1} = O_{\text{лекция}} + O_{\text{семинар}} + O_{\text{лаб. работа}} + O_{\text{ответы у доски}}$
2. Модуль №2.  $O_{\text{текущая } 2} = O_{\text{лекция}} + O_{\text{семинар}} + O_{\text{лаб. работа}} + O_{\text{ответы у доски}} + O_{\text{контр. работа}}$

Все оценки рассматриваются без округления.

Так как дисциплина преподается несколько модулей (4), то результирующая оценка учитывает текущие оценки модулей и результаты обучения на платформе Coursera (для чего студент должен получить оценку по окончании обучения). В случае если студент на завершил обучение на платформе Coursera, его оценка в этой части не учитывается и  $O_{\text{coursera}} = 0$ . Для Coursera рассматривается не округленная оценка (с точностью до сотой части балла).

Промежуточная оценка за 1 и 2 модуль вычисляется по формуле

$$O_{\text{промежуточная } 1 \text{ и } 2} = 0,2 * O_{\text{текущая } 1} + 0,4 * O_{\text{текущая } 2} + 0,2 * O_{\text{coursera}} + 0,2 * O_{\text{экзамен } 2 \text{ модуль}}$$

где  $O_{\text{текущая } 1}$ ,  $O_{\text{текущая } 2}$  – оценки текущего контроля 1, 2 модуля, без округления.

Округление производится один раз, после вычисления промежуточной оценки, по правилам арифметики.

Экзаменационная оценка не является блокирующей. Промежуточная оценка за 1 и 2 модуль не может превышать 10 баллов, в случае превышения ставится промежуточная оценка 10 баллов.

Промежуточная оценка за текущий контроль в 3 и 4 модуле учитывает результаты студента по текущему контролю следующим образом.

3. Модуль №3.  $O_{\text{текущая } 3} = O_{\text{лекция}} + O_{\text{семинар}} + O_{\text{лаб. работа}} + O_{\text{ответы у доски}} + O_{\text{контр. работа}}$
4. Модуль №4.  $O_{\text{текущая } 4} = O_{\text{лекция}} + O_{\text{семинар}} + O_{\text{лаб. работа}} + O_{\text{ответы у доски}} + O_{\text{контр. работа}}$

Промежуточная оценка за 3 и 4 модуль вычисляется по формуле

$O_{\text{промежуточная 3 и 4}} = 0,4 * O_{\text{текущая3}} + 0,4 * O_{\text{текущая 4}} + 0,2 * O_{\text{экзамен 4 модуль}}$ .

где  $O_{\text{текущая 3}}$ ,  $O_{\text{текущая 4}}$  – оценки текущего контроля 3, 4 модуля, без округления.

Округление производится один раз, после вычисления промежуточной оценки, по правилам арифметики.

Экзаменационная оценка не является блокирующей. Промежуточная оценка за 3 и 4 модуль не может превышать 10 баллов, в случае превышения ставится промежуточная оценка 10 баллов.

Результирующая оценка за дисциплину вычисляется по формуле

$O_{\text{результующая}} = 0,5 * O_{\text{промежуточная 1 и 2}} + 0,5 * O_{\text{промежуточная 3 и 4}}$ .

Округление производится по правилам арифметики.

В диплом выставляется результирующая оценка.

Информация о текущей успеваемости, а также о промежуточной, экзаменационной и результирующей оценке хранится на Яндекс-диске по адресу, который сообщается студенту на первом занятии через старост (ссылка приведена выше).

В случае, если студент изучает дисциплину по ИУП, не с начала модуля, имеются следующие особенности сдачи заданий.

1. В 1 и 2 модуле студент обязан прослушать курс на платформе Coursera и получить сертификат об окончании курса.
2. В отношении лабораторных работ, дата сдачи которых была ранее, чем был выпущен приказ о зачислении студента на образовательную программу, контрольным сроком сдачи считается конец текущего модуля. Для остальных лабораторных работ применяются общие критерии оценивания результата, включая контрольные сроки сдачи.
3. Студенты, изучающие дисциплину по ИУП, могут написать контрольную работу либо в общие сроки, либо в дополнительное время (единственная дата переписывания заранее сообщается через старост). Переписывание контрольной работы с целью повышения полученной оценки не допускается.
4. Оценки за проверочные работы на лекциях и семинарских занятиях, пропущенных студентом по любой причине, не восполняются.

Экзамен включается в расписание сессии.

На пересдаче студенту не предоставляется возможность получить дополнительный балл для компенсации оценки за текущий контроль.

При пересдаче экзамена учитывается промежуточная оценка за соответствующие модули и экзаменационная оценка, при этом результат оценивается по той же формуле, что и при сдаче экзамена. При пересдаче экзамена комиссии ею может быть принято коллегиальное решение о том, что результирующая оценка студента будет равна половине экзаменационной.

В диплом выставляется результирующая оценка.

Блокирующих элементов нет.

#### 4. Примеры оценочных средств

Оценочные средства для текущего контроля студента и оценочные средства для промежуточной аттестации расположены на сайте образовательной программы.

#### 5. Ресурсы

##### 5.1. Рекомендуемая основная литература

п/п	Наименование
1.	Ерохина Е. А., Хруслова Д. В., Журин Ю. В., Карчевский Д. О. Методические указания к лабораторному практикуму по дисциплине "Программирование (язык Паскаль)". М. : МИЭМ НИУ ВШЭ, 2014. Электронная версия данного пособия размещена на яндекс диске
2.	Ерохина Е. А., Хруслова Д. В., Журин Ю. В. Методические указания к лабораторному практикуму по дисциплине "Программирование на языке высокого уровня (язык Си)" Электронная версия данного пособия размещена на яндекс диске
3.	Ульянов, М. В. Ресурсно - эффективные компьютерные алгоритмы. Разработка и анализ: учеб. пособие для вузов / М. В. Ульянов. – М.: Физматлит, 2008. – 303 с. (51 экз)

##### 5.2. Рекомендуемая дополнительная литература

п/п	Наименование
.	Кнутт Д., Искусство программирования, <a href="http://publ.lib.ru/ARCHIVES/K/KNUT_Donal'd_Ervin/_Knut_D.E..html">http://publ.lib.ru/ARCHIVES/K/KNUT_Donal'd_Ervin/_Knut_D.E..html</a>

##### 5.3. Программное обеспечение

п/п	Наименование	Условия доступа/скачивания
	Pascal ABC Net	<i>свободное лицензионное соглашение</i>
	QtCreator	<i>свободное лицензионное соглашение</i>

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

п/п	Наименование	Условия доступа/скачивания
	Встроенный в среду	<i>Свободный доступ</i>

.	справочник по Pascal ABC Net	
.	Сайт описания библиотеки STL <a href="http://cplusplus.com">http://cplusplus.com</a> .	<i>Свободный доступ</i>
.	Qt Documentation. <a href="http://doc.qt.io/">http://doc.qt.io/</a>	<i>Свободный доступ</i>

### 5.5. Материально-техническое обеспечение дисциплины

№п/п	Наименование	Условия доступа/скачивания
1.	Курс на базе Google Classroom	<a href="https://classroom.google.com/c/MzczNTg4MjQzNDda">https://classroom.google.com/c/MzczNTg4MjQzNDda</a> Свободный доступ для всех студентов, изучающих курс
2.	Материалы курса на Яндекс диске	<a href="https://yadi.sk/d/HCYgFJgEgXq9mQ">https://yadi.sk/d/HCYgFJgEgXq9mQ</a> Свободный доступ для всех студентов, изучающих курс

При проведении лекций и семинаров используется аудитория, оборудованная проектором для отображения презентаций. Кроме того, при проведении лекций и практических занятий необходим компьютер с установленным на нем ПО для демонстрации презентаций (Power Point), а также Pascal ABC Net и Qt Creator.

Для проведения лекций, семинаров и лабораторных работ на компьютерах должен быть установлен Pascal ABC Net и Qt Creator. Для проведения лабораторных работ используются дисплейные классы (ауд. 606, 612 и 607 в МИЭМ НИУ ВШЭ).

## **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:

- 1.5.1. 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.
- 1.5.2. для лиц с нарушениями слуха: в печатной форме; в форме электронного документа; видеоматериалы с субтитрами; индивидуальные консультации с привлечением сурдопереводчика; индивидуальные задания и консультации.
- 1.5.3. для лиц с нарушениями опорно-двигательного аппарата: в печатной форме; в форме электронного документа; в форме аудиофайла; индивидуальные задания и консультации.