

Балаковский инженерно-технологический институт – филиал федерального государственного автономного образовательного учреждения высшего образования
«Национальный исследовательский ядерный университет «МИФИ»

Факультет атомной энергетики и технологий
Кафедра «Информационные системы и технологии»

РАБОЧАЯ ПРОГРАММА
по дисциплине «Языки программирования»

Направления подготовки
«09.03.02 Информационные системы и технологии»

Основная профессиональная образовательная программа
«Информационные системы и технологии»

Квалификация выпускника
Бакалавр

Форма обучения
Очная

Цель освоения дисциплины

Цель освоения дисциплины «Языки программирования» в области обучения, воспитания, развития, соотнесенные с общими целями ООП ВО: формирование культуры исследовательской и инженерной деятельности; формирование, развитие и совершенствование у студентов общепрофессиональных и цифровых универсальных компетенций в области освоения языков программирования; приобретение навыков структурного программирования на языке среднего уровня, в соответствии с трудовыми функциями профессиональных стандартов в области информационных систем, используемых в различных организациях, в том числе в организациях атомной отрасли.

Место учебной дисциплины в структуре ООП ВО

Необходимыми условиями для освоения дисциплины являются знания, умения и практические навыки по предшествующим дисциплинам и практикам:

– Информатика

Знания, умения и практические навыки, полученные при освоении дисциплины, необходимы при изучении следующих дисциплин и прохождения практик:

- Технологии программирования ;
- Инструментальные средства информационных систем ;
- Современные технологии управления базами данных;
- Архитектура информационных систем ;
- Интеллектуальные системы и технологии ;
- Объектно-ориентированное программирование ;
- Современные технологии объектно-ориентированного программирования ;
- Современные технологии интернет-программирования;
- Современные среды визуального программирования;
- Технологии обработки информации;
- Теория вероятностей и математическая статистика / Математическая статистика и прогнозирование ;
- Математическое моделирование / Численные методы;
- Мультимедиа технологии / Компьютерные технологии создания мультимедийного продукта;
- Учебная практика (ознакомительная) ;
- Учебная практика (по получению первичных профессиональных умений и навыков) ;
- Государственная итоговая аттестация.

Компетенции обучающегося, формируемые в результате освоения дисциплины

цифровые универсальные компетенции

Код компетенции	Наименование компетенции	Индикаторы достижения компетенции
УКЦ-2	Способен искать нужные источники информации и	3-УКЦ-2 Знать: методики сбора и обработки информации с использованием цифровых средств, а также актуальные российские и зарубежные источники информации в сфере про-

	данные, воспринимать, анализировать, запоминать и передавать информацию с использованием цифровых средств, а также с помощью алгоритмов при работе с полученными из различных источников данными с целью эффективного использования полученной информации для решения задач	<p>фессиональной деятельности, принципы, методы и средства решения стандартных задач профессиональной деятельности с использованием цифровых средств и с учетом основных требований информационной безопасности</p> <p>У-УКЦ-2 Уметь: применять методики поиска, сбора и обработки информации; с использованием цифровых средств, осуществлять критический анализ и синтез информации, полученной из разных источников, и решать стандартные задачи профессиональной деятельности с использованием цифровых средств и с учетом основных требований информационной безопасности</p> <p>В-УКЦ-2 Владеть: методами поиска, сбора и обработки, критического анализа и синтеза информации с использованием цифровых средств для решения поставленных задач, навыками подготовки обзоров, аннотаций, составления рефератов, научных докладов, публикаций и библиографии по научно-исследовательской работе с использованием цифровых средств и с учетом требований информационной безопасности</p>
--	---	--

общефессиональные

Код компетенции	Наименование компетенции	Индикаторы достижения компетенции
ОПК-6	Способен разрабатывать алгоритмы и программы, пригодные для практического применения в области информационных систем и технологий	<p>З-ОПК-6 Знать: языки и среды программирования; библиотеки программных модулей; шаблоны, классы объектов, используемые при разработке программного обеспечения</p> <p>У-ОПК-6 Уметь: создавать блок-схемы алгоритмов функционирования разрабатываемых программных продуктов; использовать выбранную среду программирования для написания программного кода</p> <p>В-ОПК-6 Владеть: языками и средами программирования для разработки алгоритмов и программ</p>

Задачи воспитания, реализуемые в рамках освоения дисциплины

Направление/цели	Создание условий, обеспечивающих	Использование воспитательного потенциала учебной дисциплины	Вовлечение в разноплановую внеучебную деятельность
Профессиональное воспитание	формирование культуры исследовательской и инженерной деятельности (В16)	Использование воспитательного потенциала дисциплин профессионального модуля для формирования культуры исследовательской и инженерной деятельности за счёт практических студенческих исследований современных производственных систем; проектной деятельности студентов по разработке и оптимизации технологических систем, связанной с решением реальных производственных задач; прохождения через разнообразные игровые, творческие, фестивальные формы, требующие анализа сложного объекта, постановки относительно него преобразовательных задач для их оптимального решения.	<p>1. Организация научно-практических конференций и встреч с ведущими специалистами предприятий города и ветеранами атомной отрасли.</p> <p>2. Организация и проведение предметных олимпиад и участие в конкурсах профессионального мастерства.</p>

Структура и содержание учебной дисциплины

Дисциплина изучается студентами в 1-ом и 2-ом семестрах. Общая трудоемкость дисциплины составляет 8 зачетных единиц, 288 ак. часов.

Календарный план

№ Раздела	№ Темы	Наименование раздела (темы) дисциплины	Виды учебной деятельности (час.)					Аттестация раздела (форма)	Максимальный балл за раздел
			Всего	Лекции	Лабораторные	Практические	СРС		
1 семестр									
1	Основы программирования и алгоритмизации								
	1	Понятие и эволюция языков программирования.	22	6/4*		-	16	КИ	25
	2	Основы алгоритмизации, виды и свойства алгоритмов	20	4		-	16		
3	Основы структурного программирования	38	6/3		16/2	16			
2	Основные конструкции и операторы языка программирования СИ								
	4	Основы языка программирования СИ Операторы и выражения Ввод и вывод в СИ	32	8/3	-	8/4	16	КИ	25
	5	Условия. Циклы.	32	8	-	8/4	16		
		Вид промежуточной аттестации						ЗаО	50
Итого			144	32/10		32/10	80		100
2 семестр									
3	Реализация программ с использованием языка СИ								
	1	Массивы	22	4	-	8/2	10	КИ	25
2	Указатели	22	4/2	-	8/2	10			
4	Расширенные возможности языка СИ								
	3	Многомерные массивы	22	4	-	8/2	10	КИ	25
4	Работа с функциями	33	4/4	-	8/4	16 КСР 5			
		Вид промежуточной аттестации	45					Э	50
Итого			144	16/6		32/10	46 КСР 5		100
Всего			288	48/16		64/20	126 КСР 5		

Сокращенное наименование форм текущего контроля и аттестации разделов:

Обозначение	Полное наименование
КИ	Контроль итогов
ЗаО	Зачет с оценкой
Э	Экзамен

* - занятия в интерактивной форме

Содержание лекционного курса

Тема лекции. Вопросы, отрабатываемые на лекции	Всего часов	Учебно-методическое обеспечение
1	2	3
1 семестр		
Общая классификация языков программирования. Оп-код операции. Машино-ориентированные языки 2GL. Ассемблеры. Инструкции процессора. Регистры процессора и сегментированная память. Языки программирования высокого уровня. Средства реализации языка. Интерпретатор и компилятор. Исполнимый код. Базовые типы языков программирования	6	1-6
Основы алгоритмизации, виды и свойства алгоритмов. Этапы решения задач на ЭВМ. Определение алгоритма. Словесно-формульное, графическое описание алгоритма.	4	1-6
Алфавит языка программирования, типы данных, выражения, инструкции, разделители, унарные и бинарные операторы. Организация ввода и вывода данных. Программирование алгоритмов линейной структуры на языке высокого уровня. Программирование разветвляющихся алгоритмов, условные операторы, оператор выбора Программирование циклических алгоритмов, операторы повтора с предусловием, с постусловием и заданным числом повторений.	6	1-6
Структура программы на языке Си. Объявление функций. Библиотека Си. Компилирование и выполнение программы. Программный подход к решению задач. Выражения. Оператор присваивания. Множественное присваивание. Арифметические выражения. Операторы сравнения. Логические операторы. Битовые операторы. Преобразование типов. Синтаксис оператора printf(). Модификаторы форматных команд в printf(). Синтаксис оператора scanf(). Буферизованный ввод/вывод.	8	1-6
Понятие условного выражения. Операторы выбора. Конструкция «if». Конструкция «if...else». Множественный выбор «else-if». Вложенные операторы if. Оператор switch. Циклические структуры. Цикл «for». Оператор «запятая». Вложенные «for». Цикл «while». Цикл «do... while...». Вложенные «while» и «do... while...». Операторы безусловного перехода. Оператор return. Оператор «goto». Оператор «break» и «continue». Оператор «exit()».	8	1-6
2 семестр		
Элементы и индексы массива. Определение массива. Работа с массивами в языке СИ. Инициализация массива. Строки (символьные массивы). Создание строк, строка как набор символов. Создание и способы заполнения массивов. Размерность массивов. Вычисление суммы элементов одномерного массива, поиск элементов в массиве, инверсия и сортировка элементов массива.	4	1-6
Понятие и назначение указателя. Переменные типа указатель. Операции с указателями. Присваивание значений указателям. Арифметические операции с указателями (адресная арифметика). Указатели и одномерные массивы. Указатели и многомерные массивы. Указатели и строки. Распределение памяти. Функции: free(), calloc(), realloc(), malloc().	4	1-6
Двумерные массивы. Инициализация многомерных массивов. Двумерные массивы и строки. Вычисление суммы элементов двумерного массива.	4	1-6
Аргументы функции. Возврат из функции. Тип возвращаемого значения функции. Вызов функции. Объявление функции. Прототипы функций. Переменные. Локальные переменные. Формальные переменные. Глобальные переменные.	4	1-6

Перечень практических занятий

Тема практического занятия. Задания, вопросы, отрабатываемые на практическом занятии	Всего часов	Учебно-методическое обеспечение
1	2	3
1 семестр		
Изучение интегрированной среды разработки ПО 1. Окна и панели инструментов интегрированной среды разработки ПО 2. Технология написания и отладки программного кода Программирование линейных алгоритмов 1. Разработка алгоритма вычисления математического выражения 2. Разработка программы, реализующей вычисление линейного алгоритма. Использование в программе математических операторов и методов. 3. Организация ввода/вывода в программе Программирование разветвляющихся алгоритмов 1. Разработка алгоритма и программы для решения задачи с использованием условного оператора 2. Использование оператора выбора в программировании Программирование циклических алгоритмов 1. Использование операторов повтора с предусловием при программировании циклических алгоритмов 2. Использование операторов повтора с постусловием при программировании циклических алгоритмов 3. Оператор повтора с заданным числом повторений	16	1-6
Функции ввода и вывода Си. Числовые типы данных в языке СИ. 1. Вещественные типы данных в языке СИ. 2. Символьные типы данных в языке СИ. 3. Модификаторы типов в языке СИ. 4. Арифметические операторы. 5. Бинарные операторы. 6. Битовые операторы. 7. Преобразование типов данных в СИ. 8. Операторы инкремента и декремента в языке СИ. 9. Синтаксис оператора printf(). Модификаторы форматных команд в printf(). 10. Синтаксис оператора scanf().	8	1-6
Реализация алгоритмов с разветвляющимися структурами Си. Работа с циклами в Си. 1. Синтаксис и использование оператора «switch». 2. Синтаксис и использование оператора «if». 3. Вложенные операторы «if». 4. Виды и характеристика циклических структур на языке СИ. 5. Синтаксис и использование оператора цикла «for». 6. Синтаксис и использование оператора цикла «while». 7. Синтаксис и использование оператора цикла «do while».	8	1-6
2 семестр		
Работа с массивами	8	1-6
Работа с указателями	8	1-6
Работа с многомерными массивами	8	1-6
Работа с функциями	8	1-6

Перечень лабораторных работ

Лабораторные работы Учебным планом не предусмотрены.

Задания для самостоятельной работы студентов

Задания, вопросы, для самостоятельного изучения (задания)	Всего часов	Учебно-методическое обеспечение
1	2	3
1 семестр		
Области применения языка ассемблер. Модель сегментированной памяти и её адресация в программе. Регистры общего назначения и регистры данных. Особенности и применение функциональных языков программирования.	16	1-6
Понятие и примеры использования алгоритмического языка. Описание процесса алгоритмизации решения задач. Решение задач алгоритмизации с использованием блок-схем.	16	1-6
Решение задач процедурного программирования с использованием линейных, разветвляющихся и циклических структур.	16	1-6
Назначение библиотек в Си. Запись алгоритмов решения задач с помощью псевдокодов. Взаимный приоритет групп операторов. Использование скобок в выражениях. Решение задач. Решение задач с использованием операторов ввода-вывода в Си.	16	1-6
Решение задач с использованием сложных условных выражений и оператора выбора в Си. Решение задач на использование циклической конструкции с заданным количеством итераций. Решение задач на использование циклической конструкции с предусловием. Решение задач на использование циклической конструкции с постусловием.	16	1-6
2 семестр		
Решение задач на обработку элементов одномерного двумерного массива. Выполнение практических заданий на закрепление полученных навыков.	10	1-6
Решение задач с использованием указателей. Выполнение практических заданий на закрепление полученных навыков.	10	1-6
Решение задач на обработку элементов двумерного массива. Выполнение практических заданий на закрепление полученных навыков.	10	1-6
Классы памяти. Автоматические переменные. Внешние переменные. Статические переменные. Регистровые переменные. Правила области видимости для функций. Выполнение практических заданий на закрепление полученных навыков.	21	1-6

Расчетно-графическая

Расчетно-графическая работа не предусмотрена учебным планом

Курсовая работа

Курсовая работа предусмотрена во 2 семестре. Выполняется по индивидуальному заданию, указанному в методических указаниях для выполнения КР.

Курсовая работа включает в себя задания по следующим темам: программы с линейной структурой, программы с алгоритмом ветвления, циклы, программы основанные на обработке информации хранящейся в файловых структурах. Работа выполняется на языке программирования Си в среде разработки Microsoft Visual Studio.

При подготовке и написании курсовой работы перед студентами ставятся следующие задачи:

1. Приобрести навыки самостоятельной работы с литературой по выбранной тематике;
2. Систематизировать и закрепить теоретические знания по дисциплине;

3. Развить навыки решения практических задач;
4. Уметь связно излагать материал, касающийся выбранной проблематики;
5. Самостоятельно производить необходимые виды анализа для разработки программ. Курсовая работа выполняется каждым студентом в соответствии с полученным индивидуальным заданием и представляется на рецензирование с последующей ее устной защитой и оценкой.

Целью курсовой работы является закрепление знаний и навыков полученных в процессе изучения дисциплины. Получение практических навыков по реализации программных средств на языке программирования Си.

Курсовой проект

Курсовой проект Учебным планом не предусмотрен.

Образовательные технологии

При реализации учебного материала курса используются различные образовательные технологии, способствующие созданию атмосферы свободной и творческой дискуссии как между преподавателем и студентами, так и в студенческой группе. Целью при этом является выработка у студентов навыков и компетенций, позволяющих самостоятельно вести исследовательскую и научно-педагогическую работу.

Аудиторные занятия проводятся в виде лекций с использованием ПК и компьютерного проектора, практических занятий, с использованием ПК при проведении расчетов. Самостоятельная работа студентов проводится под руководством преподавателей, с оказанием консультаций и помощи при подготовке к контрольным работам, выполнении домашних заданий.

В рамках самостоятельной работы студенты изучают электронные образовательные курсы в он-лайн формате.

Фонд оценочных средств

Фонд оценочных средств по дисциплине обеспечивает проверку освоения планируемых результатов обучения (компетенций и их индикаторов) посредством мероприятий текущего, рубежного и промежуточного контроля по дисциплине.

Связь между формируемыми компетенциями и формами контроля их освоения представлена в следующей таблице:

№ п/п	Наименование контролируемых разделов (темы)	Код и наименование индикатора достижения компетенций	Наименование оценочного средства
Входной контроль			
1	Входной контроль		Вопросы входного контроля (устно)
Аттестация разделов, текущий контроль успеваемости			
2	Основы программирования и алгоритмизации	З-ОПК-6, У-ОПК-6, В-ОПК-6, З-УКЦ-2, У-УКЦ-2, В-УКЦ-2	Контроль итогов (в форме тестирования)
3	Основные конструкции и операторы языка программирования СИ	З-ОПК-6, У-ОПК-6, В-ОПК-6, З-УКЦ-2, У-УКЦ-2, В-УКЦ-2	Контроль итогов (в форме тестирования)
4	Реализация программ с использованием языка СИ	З-ОПК-6, У-ОПК-6, В-ОПК-6, З-УКЦ-2, У-УКЦ-2, В-УКЦ-2	Контроль итогов (в форме тестирования)
5	Расширенные возможности	З-ОПК-6, У-ОПК-6, В-ОПК-6,	Контроль итогов (в форме

	языка СИ	З-УКЦ-2, У-УКЦ-2, В-УКЦ-2	тестирования)
Промежуточная аттестация			
6	Зачет с оценкой	З-ОПК-6, У-ОПК-6, В-ОПК-6, З-УКЦ-2, У-УКЦ-2, В-УКЦ-2	Вопросы к зачету с оценкой (устно)
7	Экзамен	З-ОПК-6, У-ОПК-6, В-ОПК-6, З-УКЦ-2, У-УКЦ-2, В-УКЦ-2	Вопросы к экзамену (письменно)

Входной контроль предназначен для выявления пробелов в знаниях студентов и готовности их к получению новых знаний. Оценочные средства для входного контроля представляют собой вопросы, которые задаются студентам в устной форме.

Перечень вопросов входного контроля

Вопросы входного контроля для первого семестра.

1. Дать определение информации.
2. Описать понятие "данные".
3. Дать определение информационных ресурсов.
4. В чем заключается отличие данных от информации?
5. Перечислить семейства ЭВМ.
6. Назвать виды вычислительных сетей.
7. Назвать отличия глобальных вычислительных сетей от глобальных.
8. Назвать устройства ввода и вывода информации.
9. Назовите отличие между файлом и каталогом.
10. Назовите основные операции при работе на компьютере.

Вопросы входного контроля для второго семестра.

1. Основы языка программирования СИ (история, функции, разделители, комментарии, псевдокоды).
2. Арифметические операторы в языке СИ.
3. Приоритеты операторов в языке СИ.
4. Функции ввода/вывода в языке СИ.
5. Условные операторы в языке СИ.

Текущий контроль – это непрерывно осуществляемый мониторинг уровня усвоения знаний и формирования умений и навыков в течение семестра. Текущий контроль знаний, умений и навыков студентов осуществляется в ходе учебных (аудиторных) занятий, проводимых по расписанию. Формами текущего контроля выступают опросы на практических занятиях, доклады и др.

Аттестация раздела по дисциплине проводится в форме контроля итогов в формате тестирования. Тест содержит от 10 вопросов. На выполнение задания отводится 30 минут. Тест – это форма контроля, направленная на проверку уровня освоения контролируемого теоретического и практического материала по дидактическим единицам дисциплины (терминологический аппарат, основные методы).

Примерный перечень тестовых заданий:

Тестовые задания 1. (КИ1)

1	К какому уровню языков программирования можно отнести язык СИ? а) низкого уровня б) высокого уровня с) среднего уровня
2	Машинно-зависимый язык низкого уровня в котором мнемонические имена соответствуют машинным командам называется:

	<p>a) Assembler; b) C++; c) C d) Basic e) Fortran</p>															
3	<p>В качестве разделителя программы на блоки (функции) в языке СИ используется конструкция: a) { } b) () c) [] d) start - end</p>															
4	<p>Поименованная область памяти, значение которой можно получить или изменить: a) идентификатор b) переменная c) константа d) тип данных</p>															
5	<p>Укажите тип данных, который может хранить значения в диапазоне -32768... 32767: a) char b) float c) double d) int</p>															
6	<p>Сколько существует модификаторов типов данных в языке СИ? a) два модификатора b) три модификатора c) четыре модификатора d) шесть модификаторов</p>															
7	<p>Какие из перечисленных операторов являются операторами сравнения? 1. = 2. >= 3. <= 4. ++ 5. == a) 1,2,3,4 b) 2,3,4 c) 2,3,5 d) 2,3,4,5</p>															
8	<p>Свойство алгоритма, что при точном исполнении всех предписаний процесс должен прекратиться за конечное число шагов с определенным ответом на поставленную задачу: a) понятность b) определенность c) дискретность d) результативность</p>															
9	<p>Какому логическому оператору соответствует таблица истинности?</p> <table border="1" style="margin-left: auto; margin-right: auto;"> <thead> <tr> <th>A</th> <th>B</th> <th>C</th> </tr> </thead> <tbody> <tr> <td>0</td> <td>0</td> <td>0</td> </tr> <tr> <td>1</td> <td>0</td> <td>0</td> </tr> <tr> <td>0</td> <td>1</td> <td>0</td> </tr> <tr> <td>1</td> <td>1</td> <td>1</td> </tr> </tbody> </table> <p>a) И b) ИЛИ c) НЕ d) ИЛИ-НЕ</p>	A	B	C	0	0	0	1	0	0	0	1	0	1	1	1
A	B	C														
0	0	0														
1	0	0														
0	1	0														
1	1	1														
10	<p>Верным является утверждение, что ...:</p>															

	<p>a) понятие «транслятор» является более общим по сравнению с понятием «компилятор»</p> <p>b) понятия «транслятор» и «компилятор» являются синонимами</p> <p>c) понятие «компилятор» является более общим по сравнению с понятием «транслятор»</p> <p>d) понятия «транслятор» и «компилятор» независимы друг от друга</p>
--	--

Тестовые задания 2. (КИ2)

1	<p>Укажите форматный спецификатор функции printf() в языке СИ позволяющий выводить на экран целочисленные значения:</p> <p>a) %s</p> <p>b) %c</p> <p>c) %i</p> <p>d) %l</p>
2	<p>Что произойдет в результате выполнения кода программы написанного на языке СИ?</p> <pre>int a = 10; int b = 5; if(a==b) printf("%i", a+b);</pre> <p>a) на экран выведется "a+b"</p> <p>b) на экран выведется 15</p> <p>c) на экран ничего не выведется</p> <p>d) на экран выведется 10</p>
3	<p>В результате выполнения кода написанного на языке СИ:</p> <pre>int a = 5; switch(a) { case '5': printf("Var=5"); break; case 5: printf("Var !=5"); default: printf("Unknown"); }</pre> <p>a) на экране появится «Var=5»</p> <p>b) на экране появится «Var!=5»</p> <p>c) на экране ничего не появится</p> <p>d) на экране появится «Var!=5Unknown »</p>
4	<p>Укажите цикл, тело которого выполняется один раз, не зависимо от значения выражения:</p> <p>a) do...while</p> <p>b) do</p> <p>c) foreach</p> <p>d) for</p>
5	<p>Укажите оператор который позволяет завершить работу функции, при этом управление передается в точку, из которой был произведен вызов функции:</p> <p>a) break</p> <p>b) goto</p> <p>c) return</p> <p>d) continue</p>
6	<p>Укажите функцию, которая позволяет немедленно завершить работу программы написанной на языке СИ:</p> <p>a) close()</p> <p>b) exit()</p> <p>c) getch()</p> <p>d) system()</p>

7	<p>Укажите цикл, для работы которого необходимо задать три параметра:</p> <p>a) do...while b) do c) foreach d) for</p>
8	<p>Укажите тип данных, который может хранить числа с плавающей запятой (вещественные): Укажите несколько вариантов ответа.</p> <p>a) char b) float c) double d) int</p>
9	<p>Укажите оператор, который позволяет прекратить выполнение всего цикла:</p> <p>a) break b) goto c) return d) continue</p>
10	<p>Укажите приоритет выполнения арифметических операторов (от высокого к низкому):</p> <p>1. ++ (в префиксной нотации) 2. = 3. + 4. * 5. % 6. -- (в постфиксной нотации)</p> <p>a) 1, 6, 2, 3, 4, 5 b) 4, 5, 1, 3, 6, 2 c) 6, 3, 4, 5, 1, 2 d) 1, 4, 5, 3, 2, 6</p>

Тестовые задания 3. (КИЗ)

1	<p>Как называется набор элементов данных одного типа?</p> <p>a) константа b) переменная c) массив d) список</p>
2	<p>Для массива, состоящего из N элементов, индекс последнего элемента будет равен:</p> <p>a) N-1 b) N+1 c) N d) N+2</p>
3	<p>Укажите варианты, в которых объявление массива выполнено корректно: <i>Укажите несколько вариантов.</i></p> <p>a) a[10] b) int a[10] c) int[10] a d) int a[] = {1,4,1,6} e) int a[2] = {4,8,7}</p>
4	<p>Функция strcpy(s1,s2):</p> <p>a) возвращает указатель на первый встретившийся символ s2 в строке s1 b) копирует строку из s2 в s1 c) присоединяет строку s2 к концу строки s1 d) возвращает указатель на начало первой встретившейся подстроки s2 в строке s1</p>
5	<p>Укажите функцию которая позволяет получить блок из резерва свободной памяти:</p> <p>a) malloc() b) free()</p>

	c) calloc() d) realloc()
6	Укажите функцию которая позволяет выделить дополнительное число байтов памяти: a) malloc() b) free() c) calloc() d) realloc()
7	Укажите форматный спецификатор функции printf() в языке СИ позволяющий выводить на экран числа с плавающей запятой: a) %s b) %f c) %i d) %l
8	Что выведет на экран фрагмент программы int a = 12; cout << '(' << setw(0) << a << ')';? a) (12) b) () c) (12.0)
9	Что возвращает функция, прототип которой объявлен следующим образом: struct A {char *x; int y;}; A fun(void); a) структуру как результат b) указатель на структуру c) ссылку на структуру
10	Какие библиотечные функции можно использовать для выделения и освобождения динамической памяти? a) new, free b) malloc, free c) new, delete d) malloc, delete

Тестовые задания 4. (КИ4)

1	Когда предпочтительно использовать представления объекта класса X описанием X* или X&? a) если известно, что значение указателя не будет изменяться b) если приходится перестраивать указатель на объект-элемент в течение жизни объекта-владельца c) если нужно передавать объекты в качестве параметров
2	Какие операторы можно использовать для выделения и освобождения динамической памяти? a) malloc, delete b) new, delete c) new, free d) malloc, free
3	Задан массив: int m[3][4] = { { 1, 2, 3, 4}, { 5, 6, 7, 8}, { 9, 10, 11, 12} }; Какое значение содержится в элементе m[2][3]? a) 10 b) 3 c) 7 d) 12
4	Что будет напечатано в результате выполнения фрагмента программы? char s[] = «abcd yuiο»; char *str = s; cout << str; a) ошибка выполнения b) abcd c) abcd yuiο d) синтаксическая ошибка

5	<p>Может ли виртуальная функция быть встроенной?</p> <p>a) может b) не может c) может, только если ее объявить как inline</p>
6	<p>Если имеется код <code>int x; cin >> x;</code> и вводится «1.2», то что будет в переменной x?</p> <p>a) 1 b) произойдет ошибка c) 1.2 d) другое e) 2</p>
7	<p>Укажите отличия C++ от C</p> <p>a) возможность перегрузки операторов b) поддержка обработки исключений c) поддержка пользовательских типов данных d) поддержка указателей на функции e) обобщенное программирование через шаблоны</p>
8	<p>Какая функция вызывается при аварийном завершении программы?</p> <p>a) <code>close();</code> b) <code>terminate();</code> c) <code>exit();</code></p>
9	<p>Сколько параметров может принимать catch?</p> <p>a) ни одного b) несколько c) один</p>
10	<p>Как осуществить вызов функции <code>cmp()</code> для параметров <code>int a1, double b1</code>?</p> <p>a) <code>b2 = int(b1); bool c = cmp(a1, b2);</code> b) <code>bool c = cmp(a1, b1);</code> c) <code>bool c = cmp(a1, int(b1));</code></p>

Критерии оценки тестовых заданий, устных опросов:

1. Полнота знаний теоретического контролируемого материала.
2. Количество правильных ответов.

Тестовое задание / опрос считается сданным, если студент правильно ответил на 60 процентов от общего числа вопросов.

Критерии оценивания	Оценка
Студент ответил на 90 % (и более) вопросов	Отлично
Студент ответил на 70-89 % вопросов	Хорошо
Студент ответил на 60-69 % вопросов	Удовлетворительно
Студент ответил менее чем на 59 % вопросов	Неудовлетворительно

Сумма баллов по разделам дисциплины складывается из оценок, полученных обучающимся в течение семестра по всем формам текущего контроля. Каждая форма контроля оценивается баллом в интервале от 0 до 10.

Промежуточная аттестация для первого семестра осуществляется в форме зачета с оценкой.

Перечень вопросов для подготовки к зачету с оценкой:

1. Основные понятия алгоритмизации и программирования. Виды и свойства алгоритмов.
2. Общая классификация языков программирования.

3. Машинно-ориентированные языки 2GL. Ассемблеры.
4. 2GL. Ассемблер. Регистры процессора и сегментированная память.
5. Языки программирования высокого уровня. Понятие интегрированной среды разработки.
6. Средства реализации языка. Интерпретатор и компилятор. Исполнимый код.
7. Базовые типы языков программирования. Процедурное программирование.
8. Базовые типы языков программирования. Функциональное программирование.
9. Базовые типы языков программирования. Объектно-ориентированное программирование.
10. Сравнение базовых типов языков программирования.
11. Организация ввода и вывода данных в языках высокого уровня.
12. Программирование разветвляющихся алгоритмов на ЯВУ. Понятие условного выражения. Конструкция «if». Конструкция «if...else». Множественный выбор «else-if». Вложенные операторы if.
13. Программирование разветвляющихся алгоритмов на ЯВУ. Операторы выбора.
14. Понятие циклической структуры и её виды. Программная реализация циклических управляющих структур.
15. Структурированные и неструктурированные данные. Однородные и неоднородные, статические и динамические типы данных.
16. Структура программы на языке СИ.
17. Понятие переменной. Типы данных в языке СИ.
18. Типы и характеристика операторов на языке СИ. Приоритет операторов в СИ.
19. Синтаксис оператора ввода на языке СИ
20. Синтаксис оператора вывода на языке СИ. Модификаторы форматных команд в printf());.
21. Синтаксис операторов выбора в языке СИ.
22. Виды и характеристика циклических структур на языке СИ.
23. Синтаксис и использование операторов безусловного перехода СИ.
24. Назначение и использование операторов «return» и «goto».
25. Синтаксис и использование операторов «break» и «continue».

Критерии оценки зачета с оценкой

Шкалы оценки образовательных достижений

Сумма баллов	Оценка по 4-х балльной шкале	Зачет	Оценка (ECTS)
90-100	отлично	Зачтено	A
85-89	хорошо		B
75-84			C
70-74			D
65-69	удовлетворительно		E
60-64		F	
Ниже 60	неудовлетворительно	не зачтено	F

Сумма баллов	Оценка (ECTS)	Оценка (балл за ответ на зачете)	Характеристика знаний студентов
90-100	A	Отлично	теоретическое содержание курса освоено полностью, без пробелов, необходимые практические навыки работы с освоенным материалом сформированы, все предусмотренные программой обучения учебные задания выполнены, качество их выполнения оценено числом баллов, близким к максимальному
85 - 89	B	Очень хорошо	теоретическое содержание курса освоено полностью, без пробелов, необходимые практические навыки работы с освоенным материалом в основном сформированы, все предусмотренные программой обучения учебные задания выполнены, качество выполнения большинства из них оценено числом баллов, близким к максимальному.
75 - 84	C	Хорошо	теоретическое содержание курса освоено полностью, без пробелов, некоторые практические навыки работы с освоенным материалом сформированы недостаточно, все предусмотренные программой обучения учебные задания выполнены, качество выполнения ни одного из них не оценено минимальным числом баллов, некоторые виды заданий выполнены с ошибками.
65 - 74	D	Удовлетворительно	теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые практические навыки работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий, возможно, содержат ошибки.
60-64	E	Посредственно	теоретическое содержание курса освоено частично, некоторые практические навыки работы не сформированы, многие предусмотренные программой обучения учебные задания не выполнены, либо качество выполнения некоторых из них оценено числом баллов, близким к минимальному
Ниже 60	F	Неудовлетворительно	очень слабые знания, недостаточные для понимания курса, имеется большое количество основных ошибок и недочетов

Промежуточная аттестация для второго семестра осуществляется в форме экзамена.

Перечень вопросов для подготовки к экзамену:

1. Основные понятия алгоритмизации и программирования. Виды и свойства алгоритмов.
2. Классификация языков программирования по поколениям.
3. Базовые типы языков программирования.
4. Массивы в СИ. Элементы и индексы массива.
5. Описание регулярного типа. Работа с массивами в языке СИ. Доступ к элементам массива.
6. Вычисление суммы элементов одномерного и двумерного массива, поиск элементов в массиве.

7. Работа с массивами. Алгоритмы сортировки. Инверсия и сортировка элементов массива.

8. Строки. Создание строк, строка как набор символов, обращение к символу строки. Работа со строками: конкатенация, равенство строк, поиск в строке, разделение строк, удаление и вставка символов в строке, изменение регистра символов.

9. Понятие и назначение указателей в СИ. Статическое и динамическое распределение памяти

10. Операторы * и &. Функции: malloc(), free(), calloc(), realloc().

11. Работа с файлами в СИ.

12. Синтаксис использования функций для работы с файлами. Примеры чтения и записи.

13. Модульное программирование в СИ.

14. Директивы и заголовочные файлы.

15. Структурное программирование. Принципы и примеры.

16. Пользовательские функции в СИ. Определение функций. Синтаксис.

17. Работа с функциями. Область видимости функций. Примеры реализации.

18. Входная и выходная информации функций.

19. Прототипы функций. Синтаксис, примеры использования.

20. Особенности вызова функции.

21. Передача параметров с использованием указателей. Примеры использования.

22. Передача массивов в функции. Примеры использования.

23. Передача имен функций в качестве параметров. Примеры использования.

24. Передача структур в функции. Примеры использования.

25. Рекурсия функций. Примеры использования.

Критерии оценки экзамена

Сумма баллов	Оценка (ECTS)	Оценка (балл за ответ на экзамене)	Характеристика знаний студентов
90-100	A	Отлично	теоретическое содержание курса освоено полностью, без пробелов, необходимые практические навыки работы с освоенным материалом сформированы, все предусмотренные программой обучения учебные задания выполнены, качество их выполнения оценено числом баллов, близким к максимальному
85 - 89	B	Очень хорошо	теоретическое содержание курса освоено полностью, без пробелов, необходимые практические навыки работы с освоенным материалом в основном сформированы, все предусмотренные программой обучения учебные задания выполнены, качество выполнения большинства из них оценено

			числом баллов, близким к максимальному.
75 - 84	С	Хорошо	теоретическое содержание курса освоено полностью, без пробелов, некоторые практические навыки работы с освоенным материалом сформированы недостаточно, все предусмотренные программой обучения учебные задания выполнены, качество выполнения ни одного из них не оценено минимальным числом баллов, некоторые виды заданий выполнены с ошибками.
65 - 74	Д	Удовлетворительно	теоретическое содержание курса освоено частично, но пробелы не носят существенного характера, необходимые практические навыки работы с освоенным материалом в основном сформированы, большинство предусмотренных программой обучения учебных заданий выполнено, некоторые из выполненных заданий, возможно, содержат ошибки.
60-64	Е	Посредственно	теоретическое содержание курса освоено частично, некоторые практические навыки работы не сформированы, многие предусмотренные программой обучения учебные задания не выполнены, либо качество выполнения некоторых из них оценено числом баллов, близким к минимальному
Ниже 60	Ф	Неудовлетворительно	очень слабые знания, недостаточные для понимания курса, имеется большое количество основных ошибок и недочетов

Учебно-методическое и информационное обеспечение учебной дисциплины

Основная литература:

1. Кривцов, А. Н. Алгоритмизация и программирование. Основы программирования на C/C++ : учебное пособие / А. Н. Кривцов, С. В. Хорошенко. — Санкт-Петербург : СПбГУТ им. М.А. Бонч-Бруевича, 2020. — 202 с <https://e.lanbook.com/book/180057>
2. Свердлов, С. З. Языки программирования и методы трансляции : учебное пособие для вузов / С. З. Свердлов. — 2-е изд., стер. — Санкт-Петербург : Лань, 2021. — 564 с. — ISBN 978-5-8114-8195-8.
3. Тракимус, Ю. В. Основы программирования : учебное пособие / Ю. В. Тракимус, В. П. Хиценко. — Новосибирск : НГТУ, 2020. — 66 с. — ISBN 978-5-7782-4089-6. <https://e.lanbook.com/book/152224>.

Дополнительная литература:

4. Программирование. Сборник задач: учебное пособие / О. Г. Архипов, В. С. Батасова, П. В. Гречкина [и др.] ; под редакцией М. М. Марана. — Санкт-Петербург : Лань, 2019. — 140 с. <https://e.lanbook.com/book/121485>.

5. Городняя, Л. В. Парадигма программирования : учебное пособие для вузов / Л. В. Городняя. — 2-е изд., стер. — Санкт-Петербург : Лань, 2021. — 232 с. — <https://e.lanbook.com/book/151660>. — Режим доступа: для авториз. пользователей.

Учебно-методические пособия

6. Работа со структурами в языке СИ [Текст] : метод. указ. к вып. практ. раб. по дисц. "Языки программирования" для студентов напр. подготовки ИФСТ и УПТС всех форм обуч. / сост. Михеев И. В., Виштак О.В. - Балаково: БИТИ НИЯУ МИФИ, 2018. - 16 с.

Программное обеспечение и Интернет-ресурсы:

1. Электронно-библиотечная система «Айбукс» - <https://ibooks.ru>
2. Электронно-библиотечная система «Лань» - <http://e.lanbook.com>.
3. Электронно-библиотечная система «Юрайт» - <http://www.biblio-onlain.ru>.
4. Российская национальная библиотека - <http://www.nlr.ru/>
5. Поисковая система - <http://www.yandex.ru>.
6. Гарант - <http://base.garant.ru/>.
7. Интернет-Университет Информационных Технологий - <http://www.intuit.ru>

Для проведения практических занятий и выполнения самостоятельной работы используются учебные компьютерные классы с выходом в Интернет и лицензионным программным обеспечением.

Материально-техническое обеспечение учебной дисциплины

Лекционные занятия проводятся в аудиториях: учебная мебель, учебная доска, комплект мультимедийного оборудования

Практические занятия проводятся в компьютерных классах: учебная мебель, учебная доска, комплект мультимедийного оборудования, персональные компьютеры.

Для самостоятельной работы обучающихся имеется: читальный зал с выходом в сеть Интернет: Учебная мебель, комплект мультимедийного оборудования, персональные компьютеры, МФУ.

Материально-техническое обеспечение учебной дисциплины

Лекционные занятия проводятся в аудиториях: учебная мебель, учебная доска, комплект мультимедийного оборудования

Практические занятия проводятся в компьютерных классах: учебная мебель, учебная доска, комплект мультимедийного оборудования, персональные компьютеры.

Для самостоятельной работы обучающихся имеется: читальный зал с выходом в сеть Интернет: Учебная мебель, комплект мультимедийного оборудования, персональные компьютеры, МФУ.

Учебно-методические рекомендации для студентов

1. Указания для прослушивания лекций

Перед началом занятий внимательно ознакомьтесь с учебным планом проведения лекций и списком рекомендованной литературы.

Перед посещением очередной лекции освежить в памяти основные концеп-

ции пройденного ранее материала. Подготовить при необходимости вопросы преподавателю. Не надо опасаться, что вопросы могут быть простыми.

На лекции основное внимание следует уделять не формулам и математическим выкладкам, а содержанию изучаемых вопросов, определениям и постановкам задач.

В процессе изучения лекционного курса необходимо по возможности часто возвращаться к основным понятиям и методам решения задач (здесь возможен выборочный контроль знаний студентов).

Желательно использовать конспекты лекций, в которых используется принятая преподавателем система обозначений.

Для более подробного изучения курса следует работать с рекомендованными литературными источниками и вновь появляющимися источниками.

2. Указания для участия в практических занятиях

Перед посещением уяснить тему практического занятия и самостоятельно изучить теоретические вопросы.

В конце занятия при необходимости выяснить у преподавателя неясные вопросы.

Основные результаты выполнения работы необходимо оформлять в виде бумажных отчетов.

3. Самостоятельная работа студентов обычно складывается из нескольких составляющих:

- работа с текстами: учебниками, историческими первоисточниками, дополнительной литературой, в том числе материалами интернета, а также проработка конспектов лекций;

- написание докладов, рефератов;

- подготовка к практическим занятиям;

- подготовка к зачету и экзамену непосредственно перед ним.

Таким образом, самостоятельная работа студентов является необходимым компонентом получения полноценного высшего образования.

Методические рекомендации для преподавателей

1. Указания для проведения лекций

На первой вводной лекции сделать общий обзор содержания курса и отметить новые методы и подходы к решению задач, рассматриваемых в курсе, довести до студентов требования кафедры, ответить на вопросы.

При подготовке к лекционным занятиям необходимо продумать план его проведения, содержание вступительной, основной и заключительной части лекции, ознакомиться с новинками учебной и методической литературы, публикациями периодической печати по теме лекционного занятия. Уточнить план проведения семинарского занятия по теме лекции. Перед изложением текущего лекционного материала напомнить об основных итогах, достигнутых на предыдущих лекциях. С этой целью задать несколько вопросов аудитории и осуществить выборочный контроль знания студентов.

В ходе лекционного занятия преподаватель должен назвать тему, учебные вопросы, ознакомить студентов с перечнем основной и дополнительной литературы по теме занятия. Раскрывая содержание учебных вопросов, акцентировать внимание студентов на основных категориях, явлениях и процессах, особенностях их

протекания. Раскрывать сущность и содержание различных точек зрения и научных подходов к объяснению тех или иных явлений и процессов.

Следует аргументировано обосновать собственную позицию по спорным теоретическим вопросам. Приводить примеры. Задавать по ходу изложения лекционного материала риторические вопросы и самому давать на них ответ. Это способствует активизации мыслительной деятельности студентов, повышению их внимания и интереса к материалу лекции, ее содержанию. Преподаватель должен руководить работой студентов по конспектированию лекционного материала, подчеркивать необходимость отражения в конспектах основных положений изучаемой темы, особо выделяя, категориальный аппарат. В заключительной части лекции необходимо сформулировать общие выводы по теме, раскрывающие содержание всех вопросов, поставленных в лекции. Объявить план очередного практического занятия, дать краткие рекомендации по подготовке студентов к семинару. Определить место и время консультации студентам, пожелавшим выступить на семинаре с докладами и рефератами.

На последней лекции уделить время для обзора наиболее важных положений, рассмотренных в курсе.

2. Указания для проведения практических занятий

Четко обозначить тему практического занятия.

Обсудить основные понятия, связанные с темой практического занятия.

В процессе решения задач вести дискуссию со студентами о правильности применения теоретических знаний.

Отмечать студентов, наиболее активно участвующих в решении задач и дискуссиях.

В конце практического занятия задать аудитории несколько контрольных вопросов.

3. Указания по контролю самостоятельной работы студентов

По усмотрению преподавателя задание на самостоятельную работу может быть индивидуальным или фронтальным.

При использовании индивидуальных заданий требовать от студента письменный отчет о проделанной работе.

При применении фронтальных заданий вести коллективные обсуждения со студентами основных теоретических положений.

С целью контроля качества выполнения самостоятельной работы требовать индивидуальные отчеты (допустимо вместо письменного отчета применять индивидуальные контрольные вопросы).

Программа составлена в соответствии с требованиями ОС НИЯУ МИФИ по направлению 09.03.02 Информационные системы и технологии

Рабочую программу составил

Ст. преподаватель И.В. Несытых

Рецензент: профессор О.В. Виштак

Программа одобрена на заседании УМКН «Информационные системы и технологии».

Председатель учебно-методической комиссии О.В. Виштак