МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ

Федеральное государственное бюджетное образовательное учреждение высшего образования

«ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ
УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ»
(ТУСУР)

УТВЕРЖДАЮ
Проректор по учебной работе
_____ П. Е. Троян
«___» _____ 20__ г.

РАБОЧАЯ ПРОГРАММА УЧЕБНОЙ ДИСЦИПЛИНЫ

Информационные технологии

Уровень образования: **высшее образование - бакалавриат**
Направление подготовки (специальность): **09.03.01 Информатика и вычислительная техника**
Направленность (профиль): **Интеллектуальные системы обработки информации и управления**
Форма обучения: **очная**
Факультет: **ФЭТ, Факультет электронной техники**
Кафедра: **ПрЭ, Кафедра промышленной электроники**
Курс: **1**
Семестр: **1, 2**
Учебный план набора 2018 года

Распределение рабочего времени

№	Виды учебной деятельности	1 семестр	2 семестр	Всего	Единицы
1	Лекции	26	26	52	часов
2	Практические занятия	18	18	36	часов
3	Лабораторные работы	16	16	32	часов
4	Всего аудиторных занятий	60	60	120	часов
5	Из них в интерактивной форме	12	12	24	часов
6	Самостоятельная работа	48	84	132	часов
7	Всего (без экзамена)	108	144	252	часов
8	Подготовка и сдача экзамена	36		36	часов
9	Общая трудоемкость	144	144	288	часов
		4.0	4.0	8.0	З.Е

Экзамен: 1 семестр
Дифференцированный зачет: 2 семестр

Томск 2017

Согласована на портале № 22459

Рассмотрена и одобрена на заседании кафедры
протокол № 46 от « 31 » 8 2017 г.

ЛИСТ СОГЛАСОВАНИЙ

Рабочая программа составлена с учетом требований федерального государственного образовательного стандарта высшего образования (ФГОС ВО) по направлению подготовки (специальности) 09.03.01 Информатика и вычислительная техника, утвержденного 12 января 2016 года, рассмотрена и утверждена на заседании кафедры «___» _____ 20__ года, протокол №_____.

Разработчик:

зав. кафедрой, профессор каф. ПрЭ _____ С. Г. Михальченко

Заведующий обеспечивающей каф. ПрЭ _____ С. Г. Михальченко

Рабочая программа согласована с факультетом, профилирующей и выпускающей кафедрами направления подготовки (специальности).

Декан ФЭТ _____ А. И. Воронин

Заведующий выпускающей каф. ПрЭ _____ С. Г. Михальченко

Эксперт:

Зам. зав. кафедрой ПРЭ по методической работе, профессор каф. ПРЭ _____ Н. С. Легостаев

Согласована на портале № 22459

1. Цели и задачи дисциплины

1.1. Цели дисциплины

Целью преподавания дисциплины «Информационные технологии» является изучение основ современных способов обработки информации с использованием средств вычислительной техники, знакомство с популярными программными продуктами, применяемыми как в инженерных расчетах, так в офисных технологиях.

1.2. Задачи дисциплины

– Задачи изучения дисциплины «Информационные технологии» состоят в последовательном освоении материала по основам вычислительной техники и по ее применению в решении учебных и исследовательских задач. В ходе изучения дисциплины «Информационные технологии» студенты знакомятся с устройством, принципами работы компьютера, получают представление о системном и прикладном программном обеспечении, овладевают навыками практической работы с компонентами MS Office. К задачам освоения дисциплины относится изучение пакета математических расчетов MathCAD и основ программирования на языке C++.

2. Место дисциплины в структуре ОПОП

Дисциплина «Информационные технологии» (Б1.Б.3) относится к блоку 1 (базовая часть).

Последующими дисциплинами являются: Защита выпускной квалификационной работы, включая подготовку к процедуре защиты и процедуру защиты, Математическое моделирование и программирование, Микропроцессорные устройства и системы, Объектно-ориентированное программирование, Операционные системы, Программирование и отладка микроконтроллеров, Процедурно-ориентированное программирование, Разработка сетевых приложений.

3. Требования к результатам освоения дисциплины

Процесс изучения дисциплины направлен на формирование следующих компетенций:

– ОПК-2 способностью осваивать методики использования программных средств для решения практических задач;

– ПК-1 способностью разрабатывать модели компонентов информационных систем, включая модели баз данных и модели интерфейсов "человек - электронно-вычислительная машина";

В результате изучения дисциплины студент должен:

– **знать** принципы работы персональных компьютеров, идеологию современных операционных систем, основные методы разработки алгоритмов и программ, процесс компиляции и создания программного продукта, способы размещения в памяти компьютера структур данных, используемых для представления типовых информационных объектов и обращения к ним, типовые алгоритмы обработки данных, принципы использования прикладных математических пакетов.

– **уметь** давать стратегическую оценку решаемой технической задачи, основывающуюся на понимании и ясном представлении целей исследований и абстрагировании от шаблонов и алгоритмов поиска решений; решать задачи обработки данных с помощью современных инструментальных средств конечного пользователя (MatCAD); создавать, отлаживать и тестировать программы на платформеннонезависимом языке высокого уровня (C++); представлять результаты исследований в удобном формате (MsOffice).

– **владеть** инструментами создания современных программных продуктов - отладочными средами высокого уровня; навыками написания, тестирования и отладки программных средств; навыками постановки научно-технических задач, выбора методов их решения, получения и предоставления результатов, интерпретации и анализа полученных данных.

4. Объем дисциплины и виды учебной работы

Общая трудоемкость дисциплины составляет 8.0 зачетных единицы и представлена в таблице 4.1.

Таблица 4.1 – Трудоемкость дисциплины

Виды учебной деятельности	Всего часов	Семестры	
		1 семестр	2 семестр

Согласована на портале № 22459

Аудиторные занятия (всего)	120	60	60
Лекции	52	26	26
Практические занятия	36	18	18
Лабораторные работы	32	16	16
Из них в интерактивной форме	24	12	12
Самостоятельная работа (всего)	132	48	84
Оформление отчетов по лабораторным работам	30	12	18
Проработка лекционного материала	46	16	30
Подготовка к практическим занятиям, семинарам	56	20	36
Всего (без экзамена)	252	108	144
Подготовка и сдача экзамена	36	36	
Общая трудоемкость ч	288	144	144
Зачетные Единицы	8.0	4.0	4.0

5. Содержание дисциплины

5.1. Разделы дисциплины и виды занятий

Разделы дисциплины и виды занятий приведены в таблице 5.1.

Таблица 5.1 – Разделы дисциплины и виды занятий

Названия разделов дисциплины	Лекции	Практические занятия	Лабораторные работы	Самостоятельная работа	Всего часов (без экз.)	Формируемые компетенции
1 семестр						
1 Средства обработки информационных ресурсов. Понятие алгоритма. Обращение к данным. Алгоритмические конструкции. Блок-схемы. Операторы.	6	8		7	21	ОПК-2, ПК-1
2 Обзор вычислительных математических пакетов. Принципы обращения к ячейкам памяти. Адрес переменной. Указатели.	6	2	4	11	23	ОПК-2, ПК-1
3 Подпрограммы: заголовок, описание, реализация алгоритма, вызов подпрограммы, передача параметров в тело функции.	4	2	4	12	22	ОПК-2, ПК-1
4 Массивы. Статические и динамические объекты, адресация. Одномерные и двумерные массивы.	6	2	8	11	27	ОПК-2, ПК-1
5 Возможности представления информации. Типы файлового хранения, расширения файлов. Стандартные подпрограммы обработки данных. MsOffice.	4	4		7	15	ОПК-2, ПК-1

Согласована на портале № 22459

Итого за семестр	26	18	16	48	108	
2 семестр						
6 Работа с символьными массивами. Потоковый ввод-вывод. Динамические строки.	4	4		12	20	ОПК-2, ПК-1
7 Файловая система. Программная реализация. Дескриптор файла.	6	4	4	20	34	ОПК-2, ПК-1
8 Структурированные типы данных, битовые поля, объединения. Приемы программирования микроконтроллеров, битовые операции.	6	2	8	20	36	ОПК-2, ПК-1
9 Понятие класса. Поля и методы класса. Спецификаторы доступа. Статические и динамические объекты класса.	6	4	4	18	32	ОПК-2, ПК-1
10 Доступ к данным. Информационная безопасность.	4	4		14	22	ОПК-2, ПК-1
Итого за семестр	26	18	16	84	144	
Итого	52	36	32	132	252	

5.2. Содержание разделов дисциплины (по лекциям)

Содержание разделов дисциплин (по лекциям) приведено в таблице 5.2.

Таблица 5.2 - Содержание разделов дисциплин (по лекциям)

Названия разделов	Содержание разделов дисциплины по лекциям	Трудоемкость, ч	Формируемые компетенции
1 семестр			
1 Средства обработки информационных ресурсов. Понятие алгоритма. Обращение к данным. Алгоритмические конструкции. Блок-схемы. Операторы.	Понятие информации. Предмет информатики. Информационные процессы. Информатизация общества. Компьютеры. Технология проектирования и отладки программ. Инструментальные средства программирования. Основные принципы создания программного обеспечения. Основы программирования на языке верхнего уровня (на примере С++). Процесс создания программного кода. Отладка программ. Трассировка программного кода.	2	ОПК-2, ПК-1
	Структура программы. Стандартные типы данных. Двоичный формат хранения данных. Хранение и использование чисел с плавающей точкой (мантисса, степень, знак). Вычислительная погрешность и погрешность представления данных с точки зрения разрядности процессора. Математический сопроцессор.	2	
	Алгоритмические конструкции. Операторы выбора. Перечисляемый тип данных. Операторы цикла. Использование переменных логического	2	

Согласована на портале № 22459

	типа. Оператор безусловного перехода. Организация диалога с пользователем.		
	Итого	6	
2 Обзор вычислительных математических пакетов. Принципы обращения к ячейкам памяти. Адрес переменной. Указатели.	Система компьютерной алгебры Mathematica. Пакет символьной математики MAPLE. Пакет математических и инженерных вычислений MATLAB. Программный комплекс MathCAD.	2	ОПК-2, ПК-1
	Программная реализация оперативной памяти. Диспетчер памяти. Адрес ячейки памяти, указатель. Типизированные и нетипизированные указатели. Операции "адрес" и "содержимое".	2	
	Статическое и динамическое распределение памяти. Функции динамического распределения памяти. Освобождение памяти. Операторы new и delete. Обращение к переменным "по имени" и "по адресу". Отличие указателя от динамической переменной.	2	
	Итого	6	
3 Подпрограммы: заголовок, описание, реализация алгоритма, вызов подпрограммы, передача параметров в тело функции.	Подпрограммы: процедуры и функции. Структура подпрограммы. Формальные и фактические параметры. Передача параметров в тело функции. Вызов подпрограммы. Возвращение значения, оператор return. Область видимости переменной.	2	ОПК-2, ПК-1
	Описание и вызов подпрограммы. Прототип подпрограммы. Адрес функции, указатель на подпрограмму. Область видимости переменной. Глобальные и локальные переменные. Перегрузка функций.	2	
	Итого	4	
4 Массивы. Статические и динамические объекты, адресация. Одномерные и двумерные массивы.	Массивы - хранилище однотипных данных. Одномерные (вектора) и многомерные (матрицы) массивы. Описание массива, обращение к элементам массива по индексу. Генератор случайных чисел. Работа с массивами в цикле.	2	ОПК-2, ПК-1
	Статические и динамические массивы. Функции calloc(), malloc() и free(). Адрес массива = адрес начальной ячейки. Указатель на массив. Передача массива в функцию. Динамические одномерные и двумерные массивы. Обращение к элементам массива "адрес+смещение".	2	
	Двумерные статические и динамические. Три способа размещения в памяти двумерных массивов. Освобождение памяти. Обращение к ячейкам двумерного массива.	2	
	Итого	6	
5 Возможности представления информации. Типы файлового хранения,	Операционная система. Возможности дискового хранения информации. Интерпретация данных, расширения файлов. Файловая система, файловая таблица.	2	ОПК-2, ПК-1

Согласована на портале № 22459

расширения файлов. Стандартные подпрограммы обработки данных. MsOffice.	Обращение к оборудованию ПК. BIOS. Setup BIOS. Драйверы. Работа с аудио-, видео- и графической информацией. Кодер/декодер. Приложения для работы с текстом (Word), таблицами (Excel), структурированными данными (Access) и презентациями (PowerPoint).	2	
	Итого	4	
Итого за семестр		26	
2 семестр			
6 Работа с символьными массивами. Потоковый ввод-вывод. Динамические строки.	Строки символов. Строка – массив символов. Вывод строки на экран. Последний элемент строки. Указатель на строку.	2	ОПК-2, ПК-1
	Указатель на строку. Копирование и сравнение строк. Статические и динамические строки. Функции преобразования строковых типов данных.	2	
	Итого	4	
7 Файловая система. Программная реализация. Дескриптор файла.	Файловые подсистемы ОС, хранение данных на диске. Дескриптор файла в программе. Доступ к файлу. Файловые операции (связать, открыть, закрыть, читать, писать, определить конец).	2	ОПК-2, ПК-1
	Работа с файлами при помощи потоков ввода-вывода. Библиотека . Файловые операции. Поиск в файле.	2	
	Вопросы информационной безопасности. Доступ к файлу, права доступа. Совместный доступ к информационным ресурсам, транзакция, завершенная операция.	2	
	Итого	6	
8 Структурированные типы данных, битовые поля, объединения. Приемы программирования микроконтроллеров, битовые операции.	Структуры. Описание в программе, доступ к полям структуры. Размещение в памяти. Указатели на структуру. Статические и динамические переменные структурного типа.	2	ОПК-2, ПК-1
	Массив, состоящий из структур. Структура, включающая в свой состав массивы. Битовые поля. Объединения. Статические и динамические переменные структурного типа.	2	
	Приемы программирования микроконтроллеров, битовые операции. Поразрядные логические операции. Поразрядные операции сдвига. Обращение к разрядам при помощи битовых полей.	2	
	Итого	6	
9 Понятие класса. Поля и методы класса. Спецификаторы доступа. Статические и динамические объекты	Введение в понятие "класс". Поля и методы класса. Спецификаторы доступа, использование set- и get- методов. Конструктор и деструктор.	2	ОПК-2, ПК-1
	Указатель на объект класса. Статические и динамические объекты класса. Конструктор и	2	

Согласована на портале № 22459

класса.	деструктор для статических и динамических объектов класса.		
	Отделение интерфейса программы от реализации. Дружественные функции. Перегрузка операторов. Наследование.	2	
	Итого	6	
10 Доступ к данным. Информационная безопасность.	Программный комплекс MathCAD. Структура программы. Численные и аналитические вычисления, точность вычислений. Вывод графиков. Алгоритмические возможности MathCAD. Операторы, операторные блоки. Матричные операции MathCAD. Встроенные функции.	2	ОПК-2, ПК-1
	Компьютерные сети, вопросы информационной безопасности. Локальные и глобальные компьютерные сети. Модель OSI. Адресация в сети, форматы обмена данными (сетевые протоколы). Программное обеспечение для работы в сетях. Службы сети Интернет. Угрозы информационных атак. Компьютерные вирусы.	2	
	Итого	4	
Итого за семестр		26	
Итого		52	

5.3. Разделы дисциплины и междисциплинарные связи с обеспечивающими (предыдущими) и обеспечиваемыми (последующими) дисциплинами

Разделы дисциплины и междисциплинарные связи с обеспечивающими (предыдущими) и обеспечиваемыми (последующими) дисциплинами представлены в таблице 5.3.

Таблица 5.3 - Разделы дисциплины и междисциплинарные связи

Наименование дисциплин	№ разделов данной дисциплины, для которых необходимо изучение обеспечивающих и обеспечиваемых дисциплин									
	1	2	3	4	5	6	7	8	9	10
Последующие дисциплины										
1 Защита выпускной квалификационной работы, включая подготовку к процедуре защиты и процедуру защиты	+	+	+		+					+
2 Математическое моделирование и программирование		+	+	+	+	+		+	+	
3 Микропроцессорные устройства и системы	+	+		+	+			+		
4 Объектно-ориентированное программирование	+	+	+	+		+		+	+	+
5 Операционные системы	+	+	+	+	+		+			+

Согласована на портале № 22459

6 Программирование и отладка микроконтроллеров	+					+	+			+
7 Процедурно-ориентированное программирование	+	+	+	+		+	+	+	+	
8 Разработка сетевых приложений	+	+	+	+	+		+	+	+	+

5.4. Соответствие компетенций, формируемых при изучении дисциплины, и видов занятий

Соответствие компетенций, формируемых при изучении дисциплины, и видов занятий представлено в таблице 5.4

Таблица 5.4 – Соответствие компетенций и видов занятий, формируемых при изучении дисциплины

Компетенции	Виды занятий				Формы контроля
	Лекции	Практические занятия	Лабораторные работы	Самостоятельная работа	
ОПК-2	+	+	+	+	Отчет по индивидуальному заданию, Коллоквиум, Отчет по лабораторной работе, Опрос на занятиях
ПК-1	+	+	+	+	Отчет по индивидуальному заданию, Коллоквиум, Отчет по лабораторной работе, Опрос на занятиях

6. Интерактивные методы и формы организации обучения

Технологии интерактивного обучения при разных формах занятий в часах приведены в таблице 6.1

Таблица 6.1 – Технологии интерактивного обучения при разных формах занятий в часах

Методы	Интерактивные практические занятия	Интерактивные лабораторные занятия	Интерактивные лекции	Всего
1 семестр				
Решение ситуационных задач	2	2		4
Поисковый метод	2		2	4
Исследовательский метод		2	2	4
Итого за семестр:	4	4	4	12
2 семестр				
Решение ситуационных задач	2	2		4
Поисковый метод		2	2	4
Исследовательский метод	2		2	4
Итого за семестр:	4	4	4	12

Согласована на портале № 22459

Итого	8	8	8	24

7. Лабораторные работы

Наименование лабораторных работ приведено в таблице 7.1.

Таблица 7.1 – Наименование лабораторных работ

Названия разделов	Наименование лабораторных работ	Трудоемкость, ч	Формируемые компетенции
1 семестр			
2 Обзор вычислительных математических пакетов. Принципы обращения к ячейкам памяти. Адрес переменной. Указатели.	Научиться работать со статическими и динамическими переменными. Освоить применение операций адресации и разадресации. Научиться выделять память под динамические переменные и освобождать ее при помощи операторов new и delete. Освоить обращение к динамическим объектам. Научиться выделять память под динамические переменные и освобождать ее при помощи функций calloc, malloc и free. Научиться визуализировать адреса статических и динамических объектов.	4	ОПК-2, ПК-1
	Итого	4	
3 Подпрограммы: заголовок, описание, реализация алгоритма, вызов подпрограммы, передача параметров в тело функции.	Научиться создавать подпрограммы и вызывать их из тела основной функции. Освоить применение оператора return, возвращающего значение функции. Научиться передавать параметры в функцию и корректно возвращать значения – результаты вычислений из функции. Освоить передачу параметров в функцию по значению, по ссылке и по указателю. Научиться создавать прототипы функций, освоить перегрузку функций. Изучить функции библиотеки math.h. Научиться пользоваться средствами пошаговой трассировки кода и просмотра текущих значений переменных.	4	ОПК-2, ПК-1
	Итого	4	
4 Массивы. Статические и динамические объекты, адресация. Одномерные и двумерные массивы.	Научиться создавать массивы, выделять память под элементы массива и инициировать их значениями. Научиться обращаться к элементам массивов как при помощи оператора имя[индекс], так и посредством методики *(адрес+смещение). Освоить понятия: адрес массива, адрес элемента массива, смещение, индекс. Освоить пошаговую трассировку программы с массивами, научиться отражать в окне watch элементы массива и их адреса. Научиться передавать массив в функцию.	4	ОПК-2, ПК-1
	Освоить способ динамического захвата и освобождения памяти под двумерные массивы данных (матрицы) – при помощи оператора new и посредством функций malloc( ) и calloc( ). Повторить понятия, операции и закрепить умения и навыки матричной алгебры. Уяснить практическую	4	

Согласована на портале № 22459

	разницу и сходство в языке С++ понятий «массив» и «указатель». Повысить навыки отладки программного кода на С++, трассировки программы и просмотра значений переменных в окне Watch.		
	Итого	8	
Итого за семестр		16	
2 семестр			
7 Файловая система. Программная реализация. Дескриптор файла.	Научиться работать с файловыми переменными посредстваом подпрограмм библиотек fstream.h и stdio.h. Закрепить навыки работы со строками. Научиться открывать файлы для записи и чтения, создавать и уничтожать файлы, определять конец файла.	4	ОПК-2, ПК-1
	Итого	4	
8 Структурированные типы данных, битовые поля, объединения. Приемы программирования микроконтроллеров, битовые операции.	Освоить умение работать со структурами данных. Научиться создавать новый комбинированный тип данных – структуру. Научиться обращаться к статическим и динамическим переменным этого типа и к массиву структур. Закрепить навыки работы с файлами и символьными строками.	4	ОПК-2, ПК-1
	Освоить умение работать с разрядами в байте. Научиться изменять произвольный бит в байте, не меняя остальных. Выводить на экран двоичный код переменной произвольного типа. Повысить понимание преобразований двоичного формата. Применять операции поразрядного сдвига и поразрядные логические операции, а также при помощи структур объединение (union) и битовые поля.	4	
	Итого	8	
9 Понятие класса. Поля и методы класса. Спецификаторы доступа. Статические и динамические объекты класса.	Научиться создавать класс и описывать его свойства и методы. Изучить спецификаторы доступа private и public.Применение конструктора, деструктора, set- и get- методов. Работа с объектами класса: корректно создавать, инициализировать, получать значения свойств, визуализировать и удалять. Интерфейс и реализация класса.	4	ОПК-2, ПК-1
	Итого	4	
Итого за семестр		16	
Итого		32	

8. Практические занятия (семинары)

Наименование практических занятий (семинаров) приведено в таблице 8.1.

Таблица 8. 1 – Наименование практических занятий (семинаров)

Названия разделов	Наименование практических занятий (семинаров)	Трудоемкость, ч	Формируемые компетенции

Согласована на портале № 22459

1 семестр			
1 Средства обработки информационных ресурсов. Понятие алгоритма. Обращение к данным. Алгоритмические конструкции. Блок-схемы. Операторы.	Создание проекта на Visual C++, написать простейшую программу на языке C++, отладить и запустить на выполнение программу. Научиться задавать переменные и константы следующих типов: целого, вещественного, символьного и строкового. Освоить два способа ввода с клавиатуры и вывода на экран значений этих переменных – форматный и потоковый способы. Изучить спецификаторы, форматы ввода-вывода и управляющие символы.	4	ОПК-2, ПК-1
	Изучить операторы языка C++, научиться применять их при написании программ. Научиться отслеживать значения числовых и логических переменных и выражений в процессе отладки программы. Освоить умение переводить мнемонические выражения в последовательность операторов.	4	
	Итого	8	
2 Обзор вычислительных математических пакетов. Принципы обращения к ячейкам памяти. Адрес переменной. Указатели.	Демонстрация возможностей MathCAD. Получение навыков отладки программы, выполнения вычислений и вывода результатов. Наработка умения задавать переменные, константы и графики на рабочем поле MathCAD при помощи шаблонов.	2	ОПК-2, ПК-1
	Итого	2	
3 Подпрограммы: заголовок, описание, реализация алгоритма, вызов подпрограммы, передача параметров в тело функции.	Научиться оформлять блок-схему программы в соответствии с ЕСКД. Дерево вызова процедур и список формальных параметров подпрограммы. Соотнесение фактических параметров с формальными, обращение к подпрограммам по ссылке, по значению и по указателю.	2	ОПК-2, ПК-1
	Итого	2	
4 Массивы. Статические и динамические объекты, адресация. Одномерные и двумерные массивы.	Работа с матрицами и векторами в MathCAD. Вычисление определителя матрицы A [4×4] путем разложения матрицы по соответствующему столбцу (или строке), сравнить результаты с вычислением определителя при помощи встроенного оператора.	2	ОПК-2, ПК-1
	Итого	2	
5 Возможности представления информации. Типы файлового хранения, расширения файлов. Стандартные подпрограммы обработки данных. MsOffice.	Подтверждение умений и навыков работы со следующим ПО: приложениями для работы с текстом (Word), таблицами (Excel), структурированными данными (Access) и презентациями (PowerPoint).	4	ОПК-2, ПК-1
	Итого	4	
Итого за семестр		18	

Согласована на портале № 22459

2 семестр			
6 Работа с символьными массивами. Потоковый ввод-вывод. Динамические строки.	Научиться работать со строками. Освоить способ динамического захвата и освобождения памяти под символьные массивы. Уяснить разницу между понятиями «размер массива символов» и «длина строки» при помощи функций sizeof и strlen. Освоить функции библиотеки «string.h». Научится переводить числовые данные в строковые и обратно.	4	ОПК-2, ПК-1
	Итого	4	
7 Файловая система. Программная реализация. Дескриптор файла.	Операционная система. Файловые таблицы. Обращение к оборудованию ПК. BIOS. Setup BIOS. Драйверы. Работа с аудио-, видео- и графической информацией. Кодер/декодер.	4	ОПК-2, ПК-1
	Итого	4	
8 Структурированные типы данных, битовые поля, объединения. Приемы программирования микроконтроллеров, битовые операции.	Работа с отдельными разрядами в байте. Использование объединения (union) и структуры "битовых полей". Размещение в статической и динамической памяти ячеек данных типов.	2	ОПК-2, ПК-1
	Итого	2	
9 Понятие класса. Поля и методы класса. Спецификаторы доступа. Статические и динамические объекты класса.	Создание класса и описание его свойств и методов. Спецификаторы доступа private и public. Применение конструктора, деструктора, set- и get-методов. Работа с объектами класса: корректно создавать, инициализировать, получать значения свойств, визуализировать и удалять. Выделение памяти под динамические объекты класса и освобождение ее.	4	ОПК-2, ПК-1
	Итого	4	
10 Доступ к данным. Информационная безопасность.	Информационная безопасность. Классы и их реализация. Отделение интерфейса класса от реализации класса в соответствующих *.h и *.cpp файлах, корректное подключение их к проекту, использование "препроцессорной обертки".	4	ОПК-2, ПК-1
	Итого	4	
Итого за семестр		18	
Итого		36	

9. Самостоятельная работа

Виды самостоятельной работы, трудоемкость и формируемые компетенции представлены в таблице 9.1.

Таблица 9.1 - Виды самостоятельной работы, трудоемкость и формируемые компетенции

Названия разделов	Виды самостоятельной работы	Трудоемкость, ч	Формируемые компетенции	Формы контроля
1 семестр				

Согласована на портале № 22459

1 Средства обработки информационных ресурсов. Понятие алгоритма. Обращение к данным. Алгоритмические конструкции. Блок-схемы. Операторы.	Подготовка к практическим занятиям, семинарам	4	ОПК-2, ПК-1	Опрос на занятиях, Отчет по индивидуальному заданию
	Проработка лекционного материала	3		
	Итого	7		
2 Обзор вычислительных математических пакетов. Принципы обращения к ячейкам памяти. Адрес переменной. Указатели.	Подготовка к практическим занятиям, семинарам	4	ОПК-2, ПК-1	Опрос на занятиях, Отчет по индивидуальному заданию, Отчет по лабораторной работе
	Проработка лекционного материала	3		
	Оформление отчетов по лабораторным работам	4		
	Итого	11		
3 Подпрограммы: заголовок, описание, реализация алгоритма, вызов подпрограммы, передача параметров в тело функции.	Подготовка к практическим занятиям, семинарам	4	ОПК-2, ПК-1	Опрос на занятиях, Отчет по индивидуальному заданию, Отчет по лабораторной работе
	Проработка лекционного материала	4		
	Оформление отчетов по лабораторным работам	4		
	Итого	12		
4 Массивы. Статические и динамические объекты, адресация. Одномерные и двумерные массивы.	Подготовка к практическим занятиям, семинарам	4	ОПК-2, ПК-1	Опрос на занятиях, Отчет по индивидуальному заданию, Отчет по лабораторной работе
	Проработка лекционного материала	3		
	Оформление отчетов по лабораторным работам	4		
	Итого	11		
5 Возможности представления информации. Типы файлового хранения, расширения файлов. Стандартные подпрограммы обработки данных. MsOffice.	Подготовка к практическим занятиям, семинарам	4	ОПК-2, ПК-1	Опрос на занятиях, Отчет по индивидуальному заданию
	Проработка лекционного материала	3		
	Итого	7		
Итого за семестр		48		
	Подготовка и сдача экзамена	36		Экзамен
2 семестр				
6 Работа с символьными массивами. Потоковый	Подготовка к практическим занятиям,	6	ОПК-2, ПК-1	Опрос на занятиях, Отчет

Согласована на портале № 22459

ввод-вывод. Динамические строки.	семинарам			по индивидуальному заданию
	Проработка лекционного материала	6		
	Итого	12		
7 Файловая система. Программная реализация. Дескриптор файла.	Подготовка к практическим занятиям, семинарам	8	ОПК-2, ПК-1	Опрос на занятиях, Отчет по индивидуальному заданию, Отчет по лабораторной работе
	Проработка лекционного материала	6		
	Оформление отчетов по лабораторным работам	6		
	Итого	20		
8 Структурированные типы данных, битовые поля, объединения. Приемы программирования микроконтроллеров, битовые операции.	Подготовка к практическим занятиям, семинарам	8	ОПК-2, ПК-1	Опрос на занятиях, Отчет по индивидуальному заданию, Отчет по лабораторной работе
	Проработка лекционного материала	6		
	Оформление отчетов по лабораторным работам	6		
	Итого	20		
9 Понятие класса. Поля и методы класса. Спецификаторы доступа. Статические и динамические объекты класса.	Подготовка к практическим занятиям, семинарам	6	ОПК-2, ПК-1	Опрос на занятиях, Отчет по индивидуальному заданию, Отчет по лабораторной работе
	Проработка лекционного материала	6		
	Оформление отчетов по лабораторным работам	6		
	Итого	18		
10 Доступ к данным. Информационная безопасность.	Подготовка к практическим занятиям, семинарам	8	ОПК-2, ПК-1	Опрос на занятиях, Отчет по индивидуальному заданию
	Проработка лекционного материала	6		
	Итого	14		
Итого за семестр		84		
Итого		168		

Согласована на портале № 22459

10. Курсовая работа (проект)

Не предусмотрено РУП

11. Рейтинговая система для оценки успеваемости студентов

11.1. Балльные оценки для элементов контроля

Таблица 11.1 – Балльные оценки для элементов контроля

Элементы учебной деятельности	Максимальный балл на 1-ую КТ с начала семестра	Максимальный балл за период между 1КТ и 2КТ	Максимальный балл за период между 2КТ и на конец семестра	Всего за семестр
1 семестр				
Опрос на занятиях	4	3	3	10
Отчет по индивидуальному заданию	10	10	10	30
Отчет по лабораторной работе	10	10	10	30
Итого максимум за период	24	23	23	70
Экзамен				30
Нарастающим итогом	24	47	70	100
2 семестр				
Опрос на занятиях	10	10	10	30
Отчет по индивидуальному заданию	10	10	10	30
Отчет по лабораторной работе	13	13	14	40
Итого максимум за период	33	33	34	100
Нарастающим итогом	33	66	100	100

11.2. Пересчет баллов в оценки за контрольные точки

Пересчет баллов в оценки за контрольные точки представлен в таблице 11.2.

Таблица 11. 2 – Пересчет баллов в оценки за контрольные точки

Баллы на дату контрольной точки	Оценка
≥ 90% от максимальной суммы баллов на дату КТ	5
От 70% до 89% от максимальной суммы баллов на дату КТ	4
От 60% до 69% от максимальной суммы баллов на дату КТ	3
< 60% от максимальной суммы баллов на дату КТ	2

11.3. Пересчет суммы баллов в традиционную и международную оценку

Пересчет суммы баллов в традиционную и международную оценку представлен в таблице 11.3.

Согласована на портале № 22459

Таблица 11. 3 – Пересчет суммы баллов в традиционную и международную оценку

Оценка (ГОС)	Итоговая сумма баллов, учитывает успешно сданный экзамен	Оценка (ECTS)
5 (отлично) (зачтено)	90 - 100	A (отлично)
4 (хорошо) (зачтено)	85 - 89	B (очень хорошо)
	75 - 84	C (хорошо)
	70 - 74	D (удовлетворительно)
3 (удовлетворительно) (зачтено)	65 - 69	
	60 - 64	E (посредственно)
2 (неудовлетворительно) (не зачтено)	Ниже 60 баллов	F (неудовлетворительно)

12. Учебно-методическое и информационное обеспечение дисциплины

12.1. Основная литература

1. Михальченко Сергей Геннадьевич Информационное обеспечение задач расчета и проектирования электрических схем / С. Г. Михальченко. – Томск: Изд-во Томск. гос. ун-та систем упр. и радиоэлектроники, 2016. – 200 с. : ил., табл. – Библиогр.: с. 192. [Электронный ресурс]. - http://ie.tusur.ru/docs/msg/io.rar

12.2. Дополнительная литература

1. Информатика. Базовый курс: Учебник для вузов / С. В. Симонович [и др.] ; ред. С. В. Симонович. – 2-е изд. – СПб. : Питер, 2007. – 639 с. (наличие в библиотеке ТУСУР - 57 экз.)

2. Компьютерные технологии в науке, образовании и производстве электронной техники / Кручинин В.В., Тановицкий Ю.Н., Хомич С.Л. – Томск, 2012. – 154 с. [Электронный ресурс] - Режим доступа: http://edu.tusur.ru/training/publications/967, дата обращения: 27.10.2017.

3. Саликаев, Юрий Рафаэльевич. Компьютерное моделирование и проектирование [Электронный ресурс] : учебное пособие / Ю. Р. Саликаев; Томский государственный университет систем управления и радиоэлектроники. – Электрон. текстовые дан. – Томск: [б. и.], 2012. – on-line, 94 с. [Электронный ресурс] - Режим доступа: http://edu.tusur.ru/training/publications/2548, дата обращения: 27.10.2017.

4. Михальченко С.Г. Информационные технологии. Часть 2. Профессиональные математические пакеты [Электронный ресурс]: Руководство по организации самостоятельной работы студентов / С. Г. Михальченко; Томск. гос. ун-т сист. упр. и радиоэлектроники, Кафедра промышленной электроники – Томск: ТУСУР, 2015. – 128 с. : ил., табл., прил. – Библиогр.: с. 97. [Электронный ресурс]. - http://ie.tusur.ru/docs/msg/it_2.rar

12.3 Учебно-методические пособия

12.3.1. Обязательные учебно-методические пособия

1. Михальченко С.Г. Информационные технологии. Часть 1. Пограммирование на C++ [Электронный ресурс]: Руководство по организации самостоятельной работы студентов / С. Г. Михальченко; Томск. гос. ун-т сист. упр. и радиоэлектроники, Кафедра промышленной электроники – Томск: ТУСУР, 2015. – 164 с. : ил., табл., прил. – Библиогр.: с. 162.: [Электронный ресурс]. - http://ie.tusur.ru/docs/msg/it_1.rar

2. Информационные технологии. Часть 2. Профессиональные математические пакеты. Лабораторный практикум / С. Г. Михальченко; Томский государственный университет систем управления и радиоэлектроники, Кафедра промышленной электроники – Томск: ТУСУР, 2017. – 84 с.: ил., табл., прил. – Библиогр.: с. 65. [Электронный ресурс]. - http://ie.tusur.ru/docs/msg/pmp.rar

3. Информационные технологии. Часть 1. Программирование на C++: Учебное пособие по практическим работам / С. Г. Михальченко; Томский государственный университет систем управления и радиоэлектроники, Кафедра промышленной электроники – Томск: ТУСУР, 2017. – 52 с. : ил., табл. – Библиогр.: с. 42. [Электронный ресурс]. - http://ie.tusur.ru/docs/msg/it_1.rar

Согласована на портале № 22459

12.3.2 Учебно-методические пособия для лиц с ограниченными возможностями здоровья

Учебно-методические материалы для самостоятельной и аудиторной работы обучающихся из числа инвалидов предоставляются в формах, адаптированных к ограничениям их здоровья и восприятия информации.

Для лиц с нарушениями зрения:

– в форме электронного документа;
– в печатной форме увеличенным шрифтом.

Для лиц с нарушениями слуха:

– в форме электронного документа;
– в печатной форме.

Для лиц с нарушениями опорно-двигательного аппарата:

– в форме электронного документа;
– в печатной форме.

12.4. Базы данных, информационно-справочные, поисковые системы и требуемое программное обеспечение

1. Mathematica. Система компьютерной алгебры компании Wolfram Research. Официальный сайт компании Wolfram Research http://www.wolfram.com. Способ доступа: http://www.wolfram.com/mathematica/.

2. Maple. Программный пакет компьютерной алгебры компании Waterloo Maple Inc. Официальный сайт: http://www.maplesoft.com/. Способ доступа: http://www.maplesoft.com/products/Maple/index.aspx.

3. MatLab. Пакет математических и инженерных вычислений. Официальный сайт компании-разработчика MathWorks http://www.mathworks.com/. Способ доступа: http://www.mathworks.com/products/matlab/.

4. MathCad. Система компьютерных вычислений. Официальный сайт компании-разработчика Mathsoft http://www.mathsoft.com/, в составе PTC Community http://communities.ptc.com. Способ доступа: http://www.mathcad.com/, http://communities.ptc.com/community/mathcad

13. Материально-техническое обеспечение дисциплины

13.1. Общие требования к материально-техническому обеспечению дисциплины

13.1.1. Материально-техническое обеспечение для лекционных занятий

Для проведения лекций используется учебная аудитория, расположенная по адресу 634034, Томская область, г. Томск, Вершинина улица, д. 74, 3 этаж, ауд. 204. Состав оборудования: Учебная мебель, Доска магнитно-маркерная, Компьютер класса не ниже ПЭВМ INTEL Celeron D336 2.8ГГц, подключенный к мультимедийному проектору с демонстрационным экраном. Используется лицензионное программное обеспечение, пакеты версией не ниже: Microsoft Windows XP Professional with SP3/Microsoft Windows 7 Professional with SP1; Microsoft Windows Server 2008 R2; Visual Studio 2008 EE with SP1; Microsoft Office Visio 2010; Microsoft Office Access 2003. Имеется помещения для хранения и профилактического обслуживания учебного оборудования.

13.1.2. Материально-техническое обеспечение для практических занятий

Для проведения практических (семинарских) занятий используется учебная аудитория, расположенная по адресу 634034, Томская область, г. Томск, Вершинина улица, д. 74, 3 этаж, ауд. 338, 201б или 301б. Состав оборудования: Учебная мебель, Доска магнитно-маркерная, Компьютеры класса не ниже ПЭВМ INTEL Celeron D336 2.8ГГц. - 14 шт. Используется лицензионное программное обеспечение, пакеты версией не ниже: Microsoft Windows XP Professional with SP3/Microsoft Windows 7 Professional with SP1; Microsoft Windows Server 2008 R2; Visual Studio 2008 EE with SP1; Microsoft Office Visio 2010; Microsoft Office Access 2003. Имеется помещения для хранения и профилактического обслуживания учебного оборудования.

13.1.3. Материально-техническое обеспечение для лабораторных работ

Для проведения лабораторных работ используется учебная аудитория, расположенная по

Согласована на портале № 22459

адресу 634034, Томская область, г. Томск, Вершинина улица, д. 74, 3 этаж, ауд. 338, 201б или 301б. Состав оборудования: Учебная мебель, Доска магнитно-маркерная, Компьютеры класса не ниже ПЭВМ INTEL Celeron D336 2.8ГГц. - 14 шт. Используется лицензионное программное обеспечение, пакеты версией не ниже: Microsoft Windows XP Professional with SP3/Microsoft Windows 7 Professional with SP1; Microsoft Windows Server 2008 R2; Visual Studio 2008 EE with SP1; Microsoft Office Visio 2010; Microsoft Office Access 2003. Имеется помещения для хранения и профилактического обслуживания учебного оборудования.

13.1.4. Материально-техническое обеспечение для самостоятельной работы

Требуется вычислительный класс, оснащенный персональными компьютерами, производительность которых достаточна для работы под операционной системой с установленным пакетом Microsoft Visual Studio любой версии. Количество индивидуальных рабочих мест в вычислительном классе определяет разбивку группы на подгруппы таким образом, чтобы у каждого студента имелась возможность выполнять индивидуальное задание по практике и лабораторную работу на отдельной рабочей станции. Для самостоятельной работы используется компьютерный класс, расположенный по адресу 634034, г. Томск, ул. Вершинина, 74, 1 этаж, ауд. 201б, 301б. Состав оборудования: учебная мебель; компьютеры класса не ниже ПЭВМ INTEL Celeron D336 2.8ГГц. - 14 шт.; компьютеры подключены к сети ИНТЕРНЕТ и обеспечивают доступ в электронную информационно-образовательную среду университета.

13.2. Материально-техническое обеспечение дисциплины для лиц с ограниченными возможностями здоровья

Освоение дисциплины лицами с ОВЗ осуществляется с использованием средств обучения общего и специального назначения.

При обучении студентов **с нарушениями слуха** предусмотрено использование звукоусиливающей аппаратуры, мультимедийных средств и других технических средств приема/передачи учебной информации в доступных формах для студентов с нарушениями слуха, мобильной системы обучения для студентов с инвалидностью, портативной индукционной системы. Учебная аудитория, в которой обучаются студенты с нарушением слуха, оборудована компьютерной техникой, аудиотехникой, видеотехникой, электронной доской, мультимедийной системой.

При обучении студентов **с нарушениями зрениями** предусмотрено использование в лекционных и учебных аудиториях возможности просмотра удаленных объектов (например, текста на доске или слайда на экране) при помощи видеоувеличителей для удаленного просмотра.

При обучении студентов **с нарушениями опорно-двигательного аппарата** используются альтернативные устройства ввода информации и другие технические средства приема/передачи учебной информации в доступных формах для студентов с нарушениями опорно-двигательного аппарата, мобильной системы обучения для людей с инвалидностью.

14. Фонд оценочных средств

14.1. Основные требования к фонду оценочных средств и методические рекомендации

Фонд оценочных средств и типовые контрольные задания, используемые для оценки сформированности и освоения закрепленных за дисциплиной компетенций при проведении текущей, промежуточной аттестации по дисциплине приведен в приложении к рабочей программе.

Дисциплина является обеспечивающей, формируемые компетенции требуются для освоения предметов, непосредственно относящихся к направлению "Информатика и вычислительная техника" в части программирования и информационной электроники. В связи с этим при проведении всех видов занятий и форм контроля необходимо заострять внимание обучающихся на практическом применении получаемых компетенций в предметной области будущей профессиональной деятельности.

14.2 Требования к фонду оценочных средств для лиц с ограниченными возможностями здоровья

Для студентов с инвалидностью предусмотрены дополнительные оценочные средства, перечень которых указан в таблице.

Согласована на портале № 22459

Таблица 14 – Дополнительные средства оценивания для студентов с инвалидностью

Категории студентов	Виды дополнительных оценочных средств	Формы контроля и оценки результатов обучения
С нарушениями слуха	Тесты, письменные самостоятельные работы, вопросы к зачету, контрольные работы	Преимущественно письменная проверка
С нарушениями зрения	Собеседование по вопросам к зачету, опрос по терминам	Преимущественно устная проверка (индивидуально)
С нарушениями опорно-двигательного аппарата	Решение дистанционных тестов, контрольные работы, письменные самостоятельные работы, вопросы к зачету	Преимущественно дистанционными методами
С ограничениями по общемедицинским показаниям	Тесты, письменные самостоятельные работы, вопросы к зачету, контрольные работы, устные ответы	Преимущественно проверка методами, исходя из состояния обучающегося на момент проверки

14.3 Методические рекомендации по оценочным средствам для лиц с ограниченными возможностями здоровья

Для студентов с ОВЗ предусматривается доступная форма предоставления заданий оценочных средств, а именно:

- в печатной форме;
- в печатной форме с увеличенным шрифтом;
- в форме электронного документа;
- методом чтения ассистентом задания вслух;
- предоставление задания с использованием сурдоперевода.

Студентам с инвалидностью увеличивается время на подготовку ответов на контрольные вопросы. Для таких студентов предусматривается доступная форма предоставления ответов на задания, а именно:

- письменно на бумаге;
- набор ответов на компьютере;
- набор ответов с использованием услуг ассистента;
- представление ответов устно.

Процедура оценивания результатов обучения инвалидов по дисциплине предусматривает предоставление информации в формах, адаптированных к ограничениям их здоровья и восприятия информации:

Для лиц с нарушениями зрения:

- в форме электронного документа;
- в печатной форме увеличенным шрифтом.

Для лиц с нарушениями слуха:

- в форме электронного документа;
- в печатной форме.

Для лиц с нарушениями опорно-двигательного аппарата:

- в форме электронного документа;
- в печатной форме.

При необходимости для обучающихся с инвалидностью процедура оценивания результатов обучения может проводиться в несколько этапов.

Согласована на портале № 22459

Приложение 1

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ

Федеральное государственное бюджетное образовательное учреждение высшего образования

«ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ
УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ»
(ТУСУР)

УТВЕРЖДАЮ
Проректор по учебной работе
_____ П. Е. Троян
«___» _____ 20__ г.

ФОНД ОЦЕНОЧНЫХ СРЕДСТВ ПО УЧЕБНОЙ ДИСЦИПЛИНЕ

Информационные технологии

Уровень образования: **высшее образование - бакалавриат**
Направление подготовки (специальность): **09.03.01 Информатика и вычислительная техника**
Направленность (профиль): **Интеллектуальные системы обработки информации и управления**
Форма обучения: **очная**
Факультет: **ФЭТ, Факультет электронной техники**
Кафедра: **ПрЭ, Кафедра промышленной электроники**
Курс: **1**
Семестр: **1, 2**

Учебный план набора 2018 года

Разработчик:
- зав. кафедрой, профессор каф. ПрЭ С. Г. Михальченко

Экзамен: 1 семестр
Дифференцированный зачет: 2 семестр

Томск 2017

Согласована на портале № 22459

1. Введение

Фонд оценочных средств (ФОС) является приложением к рабочей программе дисциплины (практики) и представляет собой совокупность контрольно-измерительных материалов (типовые задачи (задания), контрольные работы, тесты и др.) и методов их использования, предназначенных для измерения уровня достижения студентом установленных результатов обучения.

ФОС по дисциплине (практике) используется при проведении текущего контроля успеваемости и промежуточной аттестации студентов.

Перечень закрепленных за дисциплиной (практикой) компетенций приведен в таблице 1.

Таблица 1 – Перечень закрепленных за дисциплиной компетенций

Код	Формулировка компетенции	Этапы формирования компетенций
ОПК-2	способностью осваивать методики использования программных средств для решения практических задач	Должен знать принципы работы персональных компьютеров, идеологию современных операционных систем, основные методы разработки алгоритмов и программ, процесс компиляции и создания программного продукта, способы размещения в памяти компьютера структур данных, используемых для представления типовых информационных объектов и обращения к ним, типовые алгоритмы обработки данных, принципы использования прикладных математических пакетов.; Должен уметь давать стратегическую оценку решаемой технической задачи, основывающуюся на понимании и ясном представлении целей исследований и абстрагировании от шаблонов и алгоритмов поиска решений; решать задачи обработки данных с помощью современных инструментальных средств конечного пользователя (MatCAD); создавать, отлаживать и тестировать программы на платформеннонезависимом языке высокого уровня (C++); представлять результаты исследований в удобном формате (MsOffice).; Должен владеть инструментами создания современных программных продуктов - отладочными средами высокого уровня; навыками написания, тестирования и отладки программных средств; навыками постановки научно-технических задач, выбора методов их решения, получения и предоставления результатов, интерпретации и анализа полученных данных.;
ПК-1	способностью разрабатывать модели компонентов информационных систем, включая модели баз данных и модели интерфейсов "человек - электронно-вычислительная машина"	

Общие характеристики показателей и критериев оценивания компетенций на всех этапах приведены в таблице 2.

Согласована на портале № 22459

Таблица 2 – Общие характеристики показателей и критериев оценивания компетенций по этапам

Показатели и критерии	Знать	Уметь	Владеть
Отлично (высокий уровень)	Обладает фактическими и теоретическими знаниями в пределах изучаемой области с пониманием границ применимости	Обладает диапазоном практических умений, требуемых для развития творческих решений, абстрагирования проблем	Контролирует работу, проводит оценку, совершенствует действия работы
Хорошо (базовый уровень)	Знает факты, принципы, процессы, общие понятия в пределах изучаемой области	Обладает диапазоном практических умений, требуемых для решения определенных проблем в области исследования	Берет ответственность за завершение задач в исследовании, приспосабливает свое поведение к обстоятельствам в решении проблем
Удовлетворительно (пороговый уровень)	Обладает базовыми общими знаниями	Обладает основными умениями, требуемыми для выполнения простых задач	Работает при прямом наблюдении

2 Реализация компетенций

2.1 Компетенция ОПК-2

ОПК-2: способностью осваивать методики использования программных средств для решения практических задач.

Для формирования компетенции необходимо осуществить ряд этапов. Этапы формирования компетенции, применяемые для этого виды занятий и используемые средства оценивания представлены в таблице 3.

Таблица 3 – Этапы формирования компетенции и используемые средства оценивания

Состав	Знать	Уметь	Владеть
Содержание этапов	методики использования программных средств для решения практических задач; понимать способы реализации программных решений аппаратным комплексом информационной системы.	писать и отлаживать программы на языке C++; использовать профессиональные математические пакеты для решения практических задач; представлять полученные результаты в удобной форме стандартными средствами отображения информации.	методами написания программного кода, отладки ПО в среде разработки Visual Studio; методами и регламентами информационной безопасности; применять математические пакеты для решения практических задач.
Виды занятий	• Интерактивные практические занятия; • Интерактивные лабораторные занятия; • Интерактивные лекции; • Практические занятия;	• Интерактивные практические занятия; • Интерактивные лабораторные занятия; • Интерактивные лекции; • Практические занятия;	• Интерактивные практические занятия; • Интерактивные лабораторные занятия; • Лабораторные работы; • Самостоятельная работа;

Согласована на портале № 22459

	• Лабораторные работы; • Лекции; • Самостоятельная работа;	• Лабораторные работы; • Лекции; • Самостоятельная работа;	
Используемые средства оценивания	• Отчет по индивидуальному заданию; • Коллоквиум; • Отчет по лабораторной работе; • Опрос на занятиях; • Экзамен; • Дифференцированный зачет;	• Отчет по индивидуальному заданию; • Коллоквиум; • Отчет по лабораторной работе; • Опрос на занятиях; • Экзамен; • Дифференцированный зачет;	• Отчет по лабораторной работе; • Отчет по индивидуальному заданию; • Коллоквиум; • Экзамен; • Дифференцированный зачет;

Формулировка показателей и критериев оценивания данной компетенции приведена в таблице 4.

Таблица 4 – Показатели и критерии оценивания компетенции на этапах

Состав	Знать	Уметь	Владеть
Отлично (высокий уровень)	• Определяет, какие возможности каких программ компьютерного моделирования понадобятся для решения практических задач, оговаривает ограничения, способен предложить альтернативы; • В деталях понимает способы реализации программных решений аппаратным комплексом информационной системы, знаком с технологиями передачи информации, знает большинство стандартных протоколов и интерфейсов;	• Обладает диапазоном практических умений, требуемых для написания, отладки и тестирования программ на языке С++, способен, абстрагируясь от деталей, выбрать наилучший путь решения прикладной задачи ; • Обладает диапазоном практических умений, требуемых для использования профессиональных математических пакетов при решении практических задач, выбирает наилучший вариант; • Представляет полученные результаты в удобной форме с применением широкого спектра средств отображения информации;	• Свободно разрабатывает, отлаживает и тестирует программный код на языке С++ в наилучшей для практической задачи среде разработки; • Контролирует работу, проводит оценку, применяет методы и регламенты информационной безопасности при разработке программного обеспечения; • Владеет разными способами и прикладными математическими пакетами при решении практических задач, выбирает наиболее подходящий;
Хорошо (базовый уровень)	• Определяет, какие возможности каких программ компьютерного моделирования понадобятся для	• Способен самостоятельно написать, отладить и протестировать программу на языке С++, выбирает при этом	• Разрабатывает, отлаживает и тестирует программный код на языке С++ в среде разработки Visual Studio;

Согласована на портале № 22459

	решения практических задач; • Понимает способы реализации программных решений аппаратным комплексом информационной системы, знаком с технологиями передачи информации;	успешный путь решения прикладной задачи ; • Использует профессиональные математические пакеты для решения практических задач; • Представляет полученные результаты в удобной форме с применением основных средств отображения информации;	• Применяет методы и регламенты информационной безопасности при разработке программного обеспечения; • Владеет разными способами решения практических задач, выбирает наиболее подходящий из перечня средств MathCAD;
Удовлетворительно (пороговый уровень)	• Знает, какие возможности базовых программ компьютерного моделирования понадобятся для решения типовых задач; • Понимает основные приемы аппаратной реализации программных решений, знаком с технологиями передачи информации;	• Пользуясь справочниками и образцами способен написать, отладить и протестировать программу на языке C++, выбирает при этом типовой путь решения прикладной задачи ; • При непосредственном наблюдении использует профессиональные математические пакеты для решения типовых практических задач; • Представляет полученные результаты в требуемой форме с применением базовых средств отображения информации;	• При непосредственном наблюдении способен разработать и отладить программный код на языке C++ в среде разработки Visual Studio; • При непосредственном наблюдении применяет методы и регламенты информационной безопасности при разработке программного обеспечения; • Владеет базовыми способами решения практических задач, применяет типовые методы из перечня средств MathCAD;

2.2 Компетенция ПК-1

ПК-1: способностью разрабатывать модели компонентов информационных систем, включая модели баз данных и модели интерфейсов "человек - электронно-вычислительная машина".

Для формирования компетенции необходимо осуществить ряд этапов. Этапы формирования компетенции, применяемые для этого виды занятий и используемые средства оценивания представлены в таблице 5.

Таблица 5 – Этапы формирования компетенции и используемые средства оценивания

Состав	Знать	Уметь	Владеть
Содержание этапов	способы разработки моделей компонентов информационных систем на языке C++, включая модели баз данных (массивы, структуры, классы) и модели	применять способы разработки моделей компонентов информационных систем на языке C++, включая модели баз данных (массивы, структуры,	способами разработки моделей компонентов информационных систем на языке C++, включая модели баз данных (массивы, структуры, классы) и модели

Согласована на портале № 22459

	интерфейсов "человек - электронно-вычислительная машина" (средства ввода-вывода, файловая система, сетевые службы)	классы) и модели интерфейсов "человек - электронно-вычислительная машина" (средства ввода-вывода, файловая система, сетевые службы) при решении практических задач	интерфейсов "человек - электронно-вычислительная машина" (средства ввода-вывода, файловая система, сетевые службы)
Виды занятий	• Интерактивные практические занятия; • Интерактивные лабораторные занятия; • Интерактивные лекции; • Практические занятия; • Лабораторные работы; • Лекции; • Самостоятельная работа;	• Интерактивные практические занятия; • Интерактивные лабораторные занятия; • Интерактивные лекции; • Практические занятия; • Лабораторные работы; • Лекции; • Самостоятельная работа;	• Интерактивные практические занятия; • Интерактивные лабораторные занятия; • Лабораторные работы; • Самостоятельная работа;
Используемые средства оценивания	• Отчет по индивидуальному заданию; • Коллоквиум; • Отчет по лабораторной работе; • Опрос на занятиях; • Экзамен; • Дифференцированный зачет;	• Отчет по индивидуальному заданию; • Коллоквиум; • Отчет по лабораторной работе; • Опрос на занятиях; • Экзамен; • Дифференцированный зачет;	• Отчет по лабораторной работе; • Отчет по индивидуальному заданию; • Коллоквиум; • Экзамен; • Дифференцированный зачет;

Формулировка показателей и критериев оценивания данной компетенции приведена в таблице 6.

Таблица 6 – Показатели и критерии оценивания компетенции на этапах

Состав	Знать	Уметь	Владеть
Отлично (высокий уровень)	• Обладает фактическими и теоретическими знаниями способов разработки моделей компонентов информационных систем на языке C++, включая модели баз данных (массивы, структуры, классы) и модели интерфейсов "человек - электронно-вычислительная машина" (средства	• Применяет большинство известных способов разработки моделей компонентов информационных систем на языке C++, включая модели баз данных (массивы, структуры, классы) и модели интерфейсов "человек - электронно-вычислительная машина" (средства ввода-вывода, файловая система, сетевые	• Контролирует работу, проводит оценку, совершенствует действия по разработке моделей компонентов информационных систем на языке C++, включая модели баз данных (массивы, структуры, классы) и модели интерфейсов "человек - электронно-вычислительная машина" (средства ввода-вывода, файловая

Согласована на портале № 22459

	ввода-вывода, файловая система, сетевые службы), способен выбрать наиболее простое решение с точки зрения экономии памяти ОЗУ и времени работы;	службы) для решения практических задач;	система, сетевые службы);
Хорошо (базовый уровень)	• Обладает знаниями способов разработки моделей компонентов информационных систем на языке C++, включая модели баз данных (массивы, структуры, классы) и модели интерфейсов "человек - электронно-вычислительная машина" (средства ввода-вывода, файловая система, сетевые службы), выбирает для этого типовое решение;	• Применяет базовые способы разработки моделей компонентов информационных систем на языке C++, включая модели баз данных (массивы, структуры, классы) и модели интерфейсов "человек - электронно-вычислительная машина" (средства ввода-вывода, файловая система, сетевые службы) для решения практических задач;	• Берет ответственность за завершение задач разработки моделей компонентов информационных систем на языке C++, включая модели баз данных (массивы, структуры, классы) и модели интерфейсов "человек - электронно-вычислительная машина" (средства ввода-вывода, файловая система, сетевые службы);
Удовлетворительно (пороговый уровень)	• Обладает знаниями способов разработки моделей компонентов информационных систем на языке C++, включая модели баз данных (массивы, структуры, классы) и модели интерфейсов "человек - электронно-вычислительная машина" (средства ввода-вывода, файловая система, сетевые службы);	• При непосредственном наблюдении применяет типовые способы разработки моделей компонентов информационных систем на языке C++, включая модели баз данных (массивы, структуры, классы) и модели интерфейсов "человек - электронно-вычислительная машина" (средства ввода-вывода, файловая система, сетевые службы) при решении практических задач;	• Разрабатывает модели компонентов информационных систем на языке C++, включая модели баз данных (массивы, структуры, классы) и модели интерфейсов "человек - электронно-вычислительная машина" (средства ввода-вывода, файловая система, сетевые службы);

3 Типовые контрольные задания

Для реализации вышеперечисленных задач обучения используются типовые контрольные задания или иные материалы, необходимые для оценки знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций в процессе освоения образовательной программы, в следующем составе.

3.1 Темы коллоквиумов

– Понятие алгоритма. Алгоритмические конструкции. Блок-схемы. Операторы. Обращение к данным.

Согласована на портале № 22459

– Переменные целого, вещественного, символьного и строкового типов данных.

– Обзор вычислительных математических пакетов.

– Принципы обращения к ячейкам памяти. Адрес переменной. Указатели.

– Подпрограммы: заголовок, описание, реализация алгоритма, вызов подпрограммы, передача параметров в подпрограмму.

– Статические и динамические переменные.

– Массивы. Одномерные и двумерные массивы. Статические и динамические массивы, адресация.

– Возможности представления информации. Стандартные подпрограммы обработки данных. MsOffice.

– Работа с символьными массивами. Потоковый ввод-вывод. Динамические строки.

– Файловая система. Программная реализация. Дескриптор файла.

– Структурированные типы данных, битовые поля, объединения. Битовые операции.

– Понятие класса. Поля и методы класса. Спецификаторы доступа. Статические и динамические объекты класса.

– Доступ к данным. Информационная безопасность.

3.2 Темы индивидуальных заданий

– ЗАДАНИЕ:

– 1. Создать проект на Visual C++, написать первую программу на языке C++ – программу, выводящую на экран приветствие, отладить и запустить на выполнение программу. Продемонстрировать преподавателю выполнение программы.

– 2. Задать в соответствии с вариантом индивидуального задания переменные и присвоить им значения. Вывести на экран значения этих переменных потоковым (cout) способом. Продемонстрировать преподавателю выполнение программы.

– 3. Запрограммировать ввод заданных переменных с клавиатуры двумя способами (форматным – scanf и потоковым – cin). Вывести переменные на экран форматным способом в заданном формате. Продемонстрировать преподавателю выполнение программы.

– 4. Составить отчет, в котором отразить листинг программного кода с комментариями и привести скриншоты с результатами работы программ.

– ВАРИАНТЫ ИНДИВИДУАЛЬНЫХ ЗАДАНИЙ:

– 1. целую знаковую переменную x выводить в 5 позициях, вещественную y выводить в 3 позициях до десятичной точки и в 3 позициях после, символьную z выводить вместе с кодом, строковой w присвоить строку, содержащую вашу Фамилию И.О., номер группы, адрес и год рождения. Использовать не менее 3х различных управляющих символов.

– 2. целую беззнаковую переменную A выводить в 6 позициях, вещественную B выводить в 4 позициях до десятичной точки и в 4 позициях после, символьную C выводить вместе с кодом, строковой W присвоить строку, содержащую вашу Фамилию И.О., инициалы, город и год рождения. Использовать не менее 2х различных управляющих символов.

– 3. целую знаковую переменную P выводить в 7 позициях, вещественную R выводить в 7 позициях до десятичной точки и в 2 позициях после, символьную Q выводить вместе с кодом, строковой S присвоить строку, содержащую вашу Фамилию И.О., номер группы, рост и вес. Использовать не менее 4х различных управляющих символов.

– ЗАДАНИЕ:

– 1. Преобразовать индивидуальное задание из мнемонического описания в последовательность операторов языка C++, построить блок-схему алгоритма и доказать его адекватность текстовому заданию.

– 2. Создать проект на Visual C++, написать программу на языке C++ в соответствии с разработанным алгоритмом, отладить ее и запустить на выполнение. Продемонстрировать преподавателю выполнение программы.

– 3. Все числовые константы по заданию вводить с клавиатуры. Ввод данных организовать в диалоговом интерактивном режиме.

Согласована на портале № 22459

– 4. Выводить на экран не только конечные значения переменных, но и их текущие значения в процессе расчета.

– 5. Произвести тестирование программы, задавая, по возможности, нереальные и не предусмотренные заданием значения переменных. На этапе тестирования программы произвести проверку на зацикливание – предусмотреть невозможность попадания программы в «вечный цикл».

– 6. Составить отчет, в котором отразить листинг программного кода с комментариями, привести блок-схему алгоритма и скриншоты с результатами работы программы.

– ВАРИАНТЫ ИНДИВИДУАЛЬНЫХ ЗАДАНИЙ:

– 1.

– задать переменную i равную -10.

– задать переменную j равную 28.

– пока i меньше 23, повторять следующие действия: {

– если i равно j, то i увеличить вдвое, иначе j уменьшить на 1.

– увеличить i на 2 }.

– 2.

– задать переменную q равную -30.

– задать переменную r равную 6.

– пока q меньше или равно 20, повторять следующие действия: {

– если q равно r, то q увеличить втрое, иначе r уменьшить на 2.

– увеличить q на 3 }.

– 3.

– задать переменную M равную 28.

– задать переменную L равную 10.

– пока L меньше 19, повторять следующие действия: {

– если L не равно M, то M уменьшить на 4, иначе L увеличить вдвое.

– увеличить L на 1 }.

3.3 Темы опросов на занятиях

– Понятие информации. Предмет информатики. Информационные процессы. Информатизация общества. Компьютеры. Технология проектирования и отладки программ. Инструментальные средства программирования. Основные принципы создания программного обеспечения. Основы программирования на языке верхнего уровня (на примере C++). Процесс создания программного кода. Отладка программ. Трассировка программного кода.

– Структура программы. Стандартные типы данных. Двоичный формат хранения данных. Хранение и использование чисел с плавающей точкой (мантисса, степень, знак). Вычислительная погрешность и погрешность представления данных с точки зрения разрядности процессора. Математический сопроцессор.

– Система компьютерной алгебры Mathematica. Пакет символьной математики MAPLE. Пакет математических и инженерных вычислений MATLAB. Программный комплекс MathCAD.

– Алгоритмические конструкции. Операторы выбора. Перечисляемый тип данных. Операторы цикла. Использование переменных логического типа. Оператор безусловного перехода. Организация диалога с пользователем.

– Программная реализация оперативной памяти. Диспетчер памяти. Адрес ячейки памяти, указатель. Типизированные и нетипизированные указатели. Операции "адрес" и "содержимое". Статическое и динамическое распределение памяти. Функции динамического распределения памяти. Освобождение памяти. Операторы new и delete. Обращение к переменным "по имени" и "по адресу". Отличие указателя от динамической переменной.

– Подпрограммы: процедуры и функции. Структура подпрограммы. Формальные и фактические параметры. Передача параметров в тело функции. Вызов подпрограммы. Возвращение значения, оператор return. Область видимости переменной. Описание и вызов подпрограммы. Прототип подпрограммы. Адрес функции, указатель на подпрограмму. Область видимости переменной. Глобальные и локальные переменные. Перегрузка функций.

Согласована на портале № 22459

– Массивы - хранилище однотипных данных. Одномерные (вектора) и многомерные (матрицы) массивы. Описание массива, обращение к элементам массива по индексу. Генератор случайных чисел. Работа с массивами в цикле. Статические и динамические массивы. Функции calloc(), malloc() и free(). Адрес массива = адрес начальной ячейки. Указатель на массив. Передача массива в функцию. Динамические одномерные и двумерные массивы. Обращение к элементам массива "адрес+смещение". Двумерные статические и динамические массивы. Три способа размещения в памяти двумерных массивов. Освобождение памяти. Обращение к ячейкам двумерного массива.

– Операционная система. Возможности дискового хранения информации. Интерпретация данных, расширения файлов. Файловая система, файловая таблица. Обращение к оборудованию ПК. BIOS. Setup BIOS. Драйверы. Работа с аудио-, видео- и графической информацией. Кодер/декодер. Приложения для работы с текстом (Word), таблицами (Excel), структурированными данными (Access) и презентациями (PowerPoint).

– Строки символов. Строка – массив символов. Вывод строки на экран. Последний элемент строки. Указатель на строку. Указатель на строку. Копирование и сравнение строк. Статические и динамические строки. Функции преобразования строковых типов данных.

– Файловые подсистемы ОС, хранение данных на диске. Дескриптор файла в программе. Доступ к файлу. Файловые операции (связать, открыть, закрыть, читать, писать, определить конец). Работа с файлами при помощи потоков ввода-вывода. Поиск в файле.

– Вопросы информационной безопасности. Доступ к файлу, права доступа. Совместный доступ к информационным ресурсам, транзакция, завершенная операция.

– Структуры. Описание в программе, доступ к полям структуры. Размещение в памяти. Указатели на структуру. Статические и динамические переменные структурного типа. Массив, состоящий из структур. Структура, включающая в свой состав массивы. Битовые поля. Объединения. Статические и динамические переменные структурного типа.

– Приемы программирования микроконтроллеров, битовые операции. Поразрядные логические операции. Поразрядные операции сдвига. Обращение к разрядам при помощи битовых полей.

– Введение в понятие "класс". Поля и методы класса. Спецификаторы доступа, использование set- и get- методов. Конструктор и деструктор. Указатель на объект класса. Статические и динамические объекты класса. Конструктор и деструктор для статических и динамических объектов класса.

– Компьютерные сети, вопросы информационной безопасности. Локальные и глобальные компьютерные сети. Модель OSI. Адресация в сети, форматы обмена данными (сетевые протоколы). Программное обеспечение для работы в сетях. Службы сети Интернет. Угрозы информационных атак. Компьютерные вирусы. Отделение интерфейса программы от реализации. Дружественные функции. Перегрузка операторов. Наследование.

– Программный комплекс MathCAD. Структура программы. Численные и аналитические вычисления, точность вычислений. Вывод графиков. Алгоритмические возможности MathCAD. Операторы, операторные блоки. Матричные операции MathCAD. Встроенные функции.

3.4 Экзаменационные вопросы

– Тема "Программа на языке C++, Функции ввода-вывода"
– 1. Что представляет собой программа на языке C++?
– 2. Что такое библиотеки и как их подключить к своей программе?
– 3. Как задается переход на новую строку при выводе на экран?
– 4. Как использовать функции потокового ввода-вывода?
– 5. Как в форматном спецификаторе указать, в скольких ячейках выводить число?
– 6. Для чего нужна строка форматной спецификации, как она выглядит?
– 7. Как использовать функции форматного ввода-вывода?
– 8. Что хранится в ячейке символьного типа (char)? Как это значение выводится на экран?
– Тема "Операторы языка C++"
– 1. Чем отличаются операторы выбора от операторов цикла?
– 2. Как организовать бесконечный цикл?

Согласована на портале № 22459

– 3. Что такое «пошаговая трассировка программного кода»? Для чего она используется?
– 4. Сколько раз выполняется оператор, содержащийся в конструкции if ?
– 5. Как увидеть текущее значение переменной во время выполнения программы?
– 6. Как определить, сколько раз будет выполнен оператор, содержащийся в конструкции for?
– 7. Для чего используется оператор break? Что будет, если его применить в теле цикла?
– 8. Что такое счетчик цикла? Как его использовать в операторе for?
– 9. Какие значения могут принимать логические переменные? Можно ли их задавать числами? Какой объем памяти они занимают?
– 10. Сколько раз выполняется оператор, содержащийся в конструкции if-else ?
– 11. Каковы правила использования оператора множественного выбора switch? Для чего нужен блок default?
– 12. Чем отличается цикл с предусловием от цикла с постусловием? Приведите пример.
– 13. В чем особенность оператора continue? Как его применяют?
– 14. Что такое тернарная логическая операция? Какова семантика её применения?
– 15. Как используется оператор break? Что будет, если его пропустить в одном из case блоков оператора switch?
– Тема "Указатели и динамические переменные"
– 1. Что такое указатель и для чего он используется?
– 2. Как узнать значение переменной, если известен только ее адрес?
– 3. Какие действия выполняет оператор new?
– 4. Имеет ли статическая переменная адрес?
– 5. В чем разница между динамической переменной и указателем на нее?
– 6. Какое значение возвращают функции malloc и calloc при выделении памяти?
– 7. Как получить адрес переменной?
– 8. Какое имя можно задать динамической переменной?
– 9. Что такое адрес переменной?
– 10. Какие пять действий происходят при выделении памяти функциями malloc, calloc?
– 11. Какой размер занимает переменная-указатель?
– 12. Что такое динамическая переменная, чем она отличается от статической?
– 13. В каком случае нужно пользоваться оператором delete, а в каком – функцией free( ) для освобождения памяти?
– Тема "Подпрограммы языка C++"
– 1. Как происходит вызов подпрограммы и возвращение из нее в языке C++?
– 2. Почему нельзя изменить переменные вызывающей функции, передаваемые в вызываемую подпрограмму по значению?
– 3. Что такое перегрузка функций? Составьте пример перегруженной функции.
– 4. Как реализована передача значений переменных вызывающей функции в тело вызываемой подпрограммы по указателю? Изменяется ли значение указателя при этом?
– 5. Что такое точка возврата из функции, где она находится?
– 6. Как передать фактические параметры в подпрограмму по ссылке? Изменятся ли их значения в вызывающей программе?
– 7. Для чего нужна команда return? Какой тип данных должна иметь подпрограмма, не возвращающая никакого значения?
– 8. В чем отличие использования символа & в вызове подпрограммы и в ее описании?
– Тема "Массивы языка C++"
– 1. Что будет выведено на экран подпрограммой printf("sizeof(A)= %d", sizeof(A)), если A – одномерный статический массив double A[100]?
– 2. Как скопировать один массив в другой?
– 3. Что такое динамический массив, как выделять под него память?
– 4. Какие операторы и функции применяются для освобождения динамической памяти?
– 5. Как располагаются в памяти элементы массива?

Согласована на портале № 22459

– 6. Что будет выведено на экран подпрограммой printf("sizeof(A)= %d", sizeof(A)), если A – одномерный динамический массив double *A= new double[100]?

– 7. Что такое статический массив, какие есть два основных способа обращения к элементам массива?

– 8. Можно ли хранить в ячейках одномерного массива переменные разных типов?

– 9. Какие операторы и функции используются для выделения памяти под динамический массив?

– 10. Как определить адреса статического и динамического массивов, где они хранятся?

– 11. Являются ли эти записи идентичными для одномерного массива A: &(A[i]) или A+i?

– Тема "Двумерные массивы, матрицы"

– 1. Как размещается в памяти статический двумерный массив?

– 2 Задана матрица N×M, при такой размерности N – это число строк или столбцов?

– 3. Как выделяется память под матрицу, хранящуюся в виде одномерного массива?

– 4. Как используется вложенный цикл при работе с матрицами?

– 5. В чем состоит способ выделения памяти под матрицу как массив указателей, содержащий адреса строк матрицы.

– 6. Как организовать двумерный динамический массив символов?

– 7. Как обращаться к элементам с координатами i и j матрицы, хранящейся в виде одномерного массива?

– 8. Чем фактически являются строки любой – статической или динамической матрицы?

– 9. Как передается в подпрограмму статический двумерный массив?

– 10. Как задать статическую матрицу начальными значениями – константами?

– 11. Какие виды выделения памяти под двумерные динамические массивы вы знаете?

– 12. Как обращаться к элементам двумерного статического массива при помощи оператора «квадратные скобки» и по технологии «адрес+смещение».

– 13. Можно ли размещать в ячейках двумерного динамического массива адреса?

– 14. Как понимать следующую запись &R[0][0]? Чему равно это значение?

3.5 Темы лабораторных работ

– Стандартные типы данных, переменные, указатели. Научиться работать со статическими и динамическими переменными. Освоить применение операций адресации и разадресации. Научиться выделять память под динамические переменные и освобождать ее при помощи операторов new и delete. Освоить обращение к динамическим объектам. Научиться выделять память под динамические переменные и освобождать ее при помощи функций calloc, malloc и free. Научиться визуализировать адреса статических и динамических объектов.

– Подпрограммы. Научиться создавать подпрограммы и вызывать их из тела основной функции. Освоить применение оператора return, возвращающего значение функции. Научиться передавать параметры в функцию и корректно возвращать значения – результаты вычислений из функции. Освоить передачу параметров в функцию по значению, по ссылке и по указателю. Научиться создавать прототипы функций, освоить перегрузку функций. Изучить функции библиотеки math.h. Научиться пользоваться средствами пошаговой трассировки кода и просмотра текущих значений переменных.

– Работа с массивами. Научиться создавать массивы, выделять память под элементы массива и инициировать их значениями. Научиться обращаться к элементам массивов как при помощи оператора имя[индекс], так и посредством методики *(адрес+смещение). Освоить понятия: адрес массива, адрес элемента массива, смещение, индекс. Освоить пошаговую трассировку программы с массивами, научиться отражать в окне watch элементы массива и их адреса. Научиться передавать массив в функцию.

– Матричные операции. Освоить способ динамического захвата и освобождения памяти под двумерные массивы данных (матрицы) – при помощи оператора new и посредством функций malloc( ) и calloc( ). Повторить понятия, операции и закрепить умения и навыки матричной алгебры. Уяснить практическую разницу и сходство в языке C++ понятий «массив» и «указатель». Повысить навыки отладки программного кода на C++, трассировки программы и просмотра

Согласована на портале № 22459

значений переменных в окне Watch.

– Обращение к файлам. Научиться работать с файловыми переменными посредством подпрограмм библиотек fstream.h и stdio.h. Закрепить навыки работы со строками. Научиться открывать файлы для записи и чтения, создавать и уничтожать файлы, определять конец файла.

– Работа со структурами данных. Научиться создавать новый комбинированный тип данных – структуру. Научиться обращаться к статическим и динамическим переменным этого типа и к массиву структур. Закрепить навыки работы с файлами и символьными строками.

– Работа с разрядами в байте. Научиться изменять произвольный бит в байте, не меняя остальных. Выводить на экран двоичный код переменной произвольного типа. Повысить понимание преобразований двоичного формата. Применять операции поразрядного сдвига и поразрядные логические операции, а также при помощи структур объединение (union) и битовые поля.

– Создание класса, реализация его свойств и методов. Изучить спецификаторы доступа private и public. Применение конструктора, деструктора, set- и get- методов. Работа с объектами класса: корректно создавать, инициализировать, получать значения свойств, визуализировать и удалять. Интерфейс и реализация класса.

3.6 Вопросы дифференцированного зачета

– Тема "Классы языка С++"

– 1. Что такое класс? Для чего он используется, в чем его особенности, преимущества?

– 2. Спецификатор доступа private – для чего он применяется?

– 3. Что называется полями и методами класса? Что такое объект класса?

– 4. В чем смысл технологии отделения интерфейса от реализации методов класса?

– 5. В чем различие понятий класс и объект класса?

– 6. Может ли быть в классе несколько конструкторов, деструкторов, set- и get-методов?

– 7. Когда вызывается деструктор класса для статических и динамических объектов?

– 8. Как организовано в классах ограничение и разрешение доступа к полям и методам?

– 9. Конструктор и деструктор – для чего они нужны? Когда используются?

– 10. Спецификатор public – как он используется в конструировании класса?

– 11. Оператор принадлежности к классу («двойное двоеточие», «::») – для чего и как он применяется? Можно ли обойтись без него?

– 12. Что такое set- и get- методы? В каком случае они необходимы?

– 13. Как реализовано обращение к полям и методам статических и динамических объектов класса?

– 14. Что такое унарные и бинарные операторы? Что такое приоритет операций?

– Тема "Вопросы информационной безопасности"

– 15. Какие спецификаторы доступа вы знаете? В чем их особенность?

– 16. Что такое заголовочный файл и чем он отличается от файла реализации методов? Как их использовать в проекте?

– 17. Можно прописать реализацию методов класса в самом теле класса, а можно – в другом месте и даже в другом файле. В чем различие?

– 18. Что такое спецификатор доступа friend? Как он используется?

– 19. Что представляют собой операторы? Можно ли создавать операторы для собственных классов?

– 20. Что такое «препроцессорная обертка»? Как это применяется?

– Тема "Файловые операции в языке С++"

– 1. Что такое текстовый файл? Опишите последовательность записи строки в файл потоковым способом.

– 2. Что представляют собой файловые переменные типа FILE из библиотеки <stdio.h> как с ними работать? Какое им присваивается значение?

– 3. С точки зрения механизма размещения двоичных данных в памяти компьютера имеются ли особенности хранения файлов различного назначения?

– 4. Опишите логику последовательности чтения отдельного слова из файла для файловых

Согласована на портале № 22459

переменных (FILE) из библиотеки <stdio.h>.

– 5. Что представляют собой объекты файлового ввода-вывода <ifstream> и <ofstream> из библиотеки <iostream>? Как с ними работать?

– 6. Опишите последовательность записи числа в файл форматным способом, используя переменные типа FILE из библиотеки <stdio.h>.

– 7. В чем разница использования метода getline() и оператора >> при потоковом вводе данных из файла – подпрограмм класса <ifstream>.

– 8. По какому признаку можно определить, как будут интерпретированы операционной системой данные, хранящиеся в файле?

– 9. Как указать для чего – для чтения или записи открывается файл? Объяснение привести для переменных типа FILE из библиотеки <stdio.h> и объектов потоковых классов.

– Тема "Структурированные типы данных"

– 1. Возможно ли в полях структуры размещать данные различных типов?

– 2. Как выделяется память под динамическую переменную структурного типа, как освобождается?

– 3. Как размещаются в переменной структурного типа ячейки полей структуры?

– 4. Как осуществляется обращение к полям статической переменной структурного типа?

– 5. Что такое объединение (union), как размещаются поля в переменных такого типа данных?

– 6. Что такое указатель на структуру, как с ним работать, как обращаться к полям структурной переменной «по адресу?»

– 7. Можно ли указывать, сколько бит выделять под каждое поле структуры?

– 8. Как осуществляется обращение к полям динамической переменной структурного типа?

– 9. Как передается в подпрограмму статическая переменная структурного типа?

– Тема "Битовые операции в языке C++"

– 1. Как запрограммировать побитное обращение к переменным целых типов.

– 2. Как работает двоичная побитная операция & «и», что получится в результате вычисления 23&126?

– 3. Как размещаются в памяти поля переменной типа union?

– 4. Как работает двоичная побитная операция | «или», что получится в результате вычисления 23|112?

– 5. Как работает двоичная побитная операция ^ «исключающее или», что получится в результате 103^112?

– 6. Объединение содержит три поля unsigned short X, double Z и char Y, какой размер будет занимать переменная этого типа?

– 7. Как работает побитная операция «сдвиг», что получится в результате 23>>2?

– 8. Что такое битовые поля и как с ними работать?

– Тема "Символьные массивы, строки"

– 1. Что представляет собой символьная строка?

– 2. Как используются стандартные функции форматного и потокового ввода-вывода в работе со строками?

– 3. Размер символьного массива и длина строки – это одно и то же?

– 4. Как обозначается конец строки, для чего он нужен?

– 5. Какие операторы и функции служат для освобождения памяти, занятой динамическим символьным массивом?

– 6. Какие стандартные библиотеки для работы со строками вы знаете? Можно ли программировать работу со строками без них?

– 7. Как производится присваивание (копирование) строковых переменных?

– 8. Может ли строка символов быть статической или динамической?

– 9. Как определить количество символов в массиве? В строке?

– 10. Какие символы строки будут выводиться на экран стандартными подпрограммами

Согласована на портале № 22459

ввода-вывода, а какие – нет?

– 11. Можно ли обращаться к элементам строки при помощи оператора «квадратные скобки», а по технологии «адрес + смещение»?

– 12. Какие операторы и функции служат для выделения памяти под динамический символьный массив?

– 13. Что такое адрес строки? Как сравнить две строковых переменных?

– 14. Какие функции служат для перевода числа в символьную строку, какие – для перевода символьной строки в число? Что возвращается, если перевод сделать не удалось?

4 Методические материалы

Для обеспечения процесса обучения и решения задач обучения используются следующие материалы:

– методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы фор-мирования компетенций, согласно п. 12 рабочей программы.

4.1. Основная литература

1. Михальченко Сергей Геннадьевич Информационное обеспечение задач расчета и проектирования электрических схем / С. Г. Михальченко. – Томск: Изд-во Томск. гос. ун-та систем упр. и радиоэлектроники, 2016. – 200 с. : ил., табл. – Библиогр.: с. 192. [Электронный ресурс]. - http://ie.tusur.ru/docs/msg/io.rar

4.2. Дополнительная литература

1. Информатика. Базовый курс: Учебник для вузов / С. В. Симонович [и др.] ; ред. С. В. Симонович. – 2-е изд. – СПб. : Питер, 2007. – 639 с. (наличие в библиотеке ТУСУР - 57 экз.)

2. Компьютерные технологии в науке, образовании и производстве электронной техники / Кручинин В.В., Тановицкий Ю.Н., Хомич С.Л. – Томск, 2012. – 154 с. [Электронный ресурс] - Режим доступа: http://edu.tusur.ru/training/publications/967, свободный.

3. Саликаев, Юрий Рафаэльевич. Компьютерное моделирование и проектирование [Электронный ресурс] : учебное пособие / Ю. Р. Саликаев; Томский государственный университет систем управления и радиоэлектроники. – Электрон. текстовые дан. – Томск: [б. и.], 2012. – on-line, 94 с. [Электронный ресурс] - Режим доступа: http://edu.tusur.ru/training/publications/2548, свободный.

4. Михальченко С.Г. Информационные технологии. Часть 2. Профессиональные математические пакеты [Электронный ресурс]: Руководство по организации самостоятельной работы студентов / С. Г. Михальченко; Томск. гос. ун-т сист. упр. и радиоэлектроники, Кафедра промышленной электроники – Томск: ТУСУР, 2015. – 128 с. : ил., табл., прил. – Библиогр.: с. 97. [Электронный ресурс]. - http://ie.tusur.ru/docs/msg/it_2.rar

4.3. Обязательные учебно-методические пособия

1. Михальченко С.Г. Информационные технологии. Часть 1. Пограммирование на С++ [Электронный ресурс]: Руководство по организации самостоятельной работы студентов / С. Г. Михальченко; Томск. гос. ун-т сист. упр. и радиоэлектроники, Кафедра промышленной электроники – Томск: ТУСУР, 2015. – 164 с. : ил., табл., прил. – Библиогр.: с. 162.: [Электронный ресурс]. - http://ie.tusur.ru/docs/msg/it_1.rar

2. Информационные технологии. Часть 2. Профессиональные математические пакеты. Лабораторный практикум / С. Г. Михальченко; Томский государственный университет систем управления и радиоэлектроники, Кафедра промышленной электроники – Томск: ТУСУР, 2017. – 84 с.: ил., табл., прил. – Библиогр.: с. 65. [Электронный ресурс]. - http://ie.tusur.ru/docs/msg/pmp.rar

3. Информационные технологии. Часть 1. Программирование на С++: Учебное пособие по практическим работам / С. Г. Михальченко; Томский государственный университет систем управления и радиоэлектроники, Кафедра промышленной электроники – Томск: ТУСУР, 2017. – 52 с. : ил., табл. – Библиогр.: с. 42. [Электронный ресурс]. - http://ie.tusur.ru/docs/msg/it_1.rar

4.4. Базы данных, информационно справочные и поисковые системы

1. Mathematica. Система компьютерной алгебры компании Wolfram Research. Официальный сайт компании Wolfram Research http://www.wolfram.com. Способ доступа:

Согласована на портале № 22459

http://www.wolfram.com/mathematica/ .

2. Maple. Программный пакет компьютерной алгебры компании Waterloo Maple Inc. Официальный сайт: http://www.maplesoft.com/. Способ доступа: http://www.maplesoft.com/products/Maple/index.aspx .

3. MatLab. Пакет математических и инженерных вычислений. Официальный сайт компании-разработчика MathWorks http://www.mathworks.com/. Способ доступа: http://www.mathworks.com/products/matlab/ .

4. MathCad. Система компьютерных вычислений. Официальный сайт компании-разработчика Mathsoft http://www.mathsoft.com/, в составе PTC Community http://communities.ptc.com. Способ доступа: http://www.mathcad.com/, http://communities.ptc.com/community/mathcad

Согласована на портале № 22459