МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
Федеральное государственное бюджетное образовательное учреждение высшего образования
«ИВАНОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»
Шуйский филиал ИвГУ

Кафедра математики, информатики и методики обучения

ОДОБРЕНО:

Руководитель ОП

_____ С.В. Белов
(подпись)

«21»февраля 2023 г.

Рабочая программа дисциплины
СИСТЕМНОЕ ПРОГРАММИРОВАНИЕ

Уровень высшего образования:	бакалавриат
Квалификация выпускника:	бакалавр
Направление подготовки:	44.03.01 Педагогическое образование
Направленность (профиль) образовательной программы:	Информационные технологии и робототехника

Шуя

1. Цели освоения дисциплины содействовать становлению профессиональной компетентности педагога через формирование целостного представления о системном программировании и развитие у студентов практических навыков работы со средствами языка С++, освоение студентами методики построения программ, работающих на низком уровне.

2. Место дисциплины в структуре ОП

Дисциплина «Системное программирование» является дисциплиной по выбору, относится к формируемой участниками образовательных отношений части блока дисциплин Б1, входит в предметно-методический модуль.

Успешное освоение данной дисциплины будет способствовать готовности обучающихся к освоению дисциплин: «Логическое программирование», «Основы искусственного интеллекта»; успешному прохождению следующих практик: «Производственная практика, преподавательская практика (Предметно-методический модуль)», «Производственная практика, педагогическая (комплексная) практика (Предметно-методический модуль)», «Производственная практика, научно-исследовательская работа (Предметно-методический модуль)», подготовке к сдаче и сдаче государственного экзамена, выполнению и защите выпускной квалификационной работы.

Студент, приступающий к изучению данной дисциплины, должен:

Знать:

- особенности системного и критического мышления;
- основы современных технологий сбора, обработки и представления информации;
- основные понятия и термины, связанные с алгоритмизацией и программированием.

Уметь:

- анализировать источники информации с точки зрения временных и пространственных условий их возникновения;
- получать, обрабатывать и анализировать информацию из различных источников;
- использовать современные информационно-коммуникационные технологии (включая пакеты прикладных программ, локальные и глобальные компьютерные сети) для сбора, обработки и анализа информации;
- работать с математической и технической литературой;
- использовать программную поддержку курса и оценивать ее методическую целесообразность.

Владеть:

- навыками сопоставления разных источников информации с целью выявления их противоречий и поиска достоверных суждений;
- основными методами математической обработки информации;
- методами математического доказательства гипотез;
- базовыми понятиями и терминами в области математики и информатики;

3. Перечень планируемых результатов обучения по дисциплине, соотнесенных с планируемыми результатами освоения образовательной программы

3.1. Компетенции, формированию которых способствует данная дисциплина

При реализации дисциплины формируются следующие компетенции в соответствии с ФГОС ВО по данному направлению подготовки:

а) универсальные компетенции (УК) и индикаторы их достижения:

Наименование категории (группы) универсальных компетенций	Код и наименование универсальной компетенции выпускника	Код и наименование индикатора достижения универсальной компетенции
Системное и критическое мышление	УК-1. Способен осуществлять поиск, критический анализ и синтез информации, применять системный подход для решения поставленных задач	УК-1.1. Демонстрирует знание особенностей системного и критического мышления и готовность к нему. УК-1.2. Применяет логические формы и процедуры, способен к рефлексии по поводу собственной и чужой мыслительной деятельности. УК-1.3. Анализирует источник информации с точки зрения временных и пространственных условий его возникновения. УК-1.4. Анализирует ранее сложившиеся в науке оценки информации. УК-1.5. Сопоставляет разные источники информации с целью выявления их противоречий и поиска достоверных суждений. УК-1.6. Аргументировано формирует собственное суждение и оценку информации, принимает обоснованное решение. УК-1.7. Определяет практические последствия предложенного решения задачи.

б) общепрофессиональные компетенции (ОПК) и индикаторы их достижения:

–

в) профессиональные компетенции (ПК) и индикаторы их достижения:

Код и наименование профессиональной компетенции выпускника	Код и наименование индикатора достижения профессиональной компетенции	Основание (ПС, анализ опыта)
Тип задач профессиональной деятельности: педагогический		
ПК-6. Способен выделять структурные элементы, входящие в систему познания предметной области (в соответствии с профилем и уровнем обучения), анализировать их в единстве содержания, формы	ПК-6.1. Выделяет и анализирует единицы различных уровней знаковой математической структуры в единстве их содержания, формы и функций. ПК-6.2. Выделяет и анализирует закономерности и явления информатики как науки в их структурном и функциональном единстве с информационно-коммуникационными технологиями.	Профессиональный стандарт 01.001 «Педагог (педагогическая деятельность в сфере дошкольного, начального общего, основного общего, среднего общего образования) (воспита-

и выполняемых функций	ПК-6.3. Знает и умеет анализировать аксиомы, определения и математические доказательства в единстве с прикладными аспектами использования.	тель, учитель)»
ПК-7. Способен определять собственную позицию относительно дискуссион ных проблем предметной области (в соответствии с профилем и уровнем обучения)	ПК-7.1. Проявляет мировоззренческую рефлексию при анализе проблем и тенденций в области информационных технологий и робототехники. ПК-7.2. Проявляет способность аргументированно, логические верно и ясно выражать свою позицию по обсуждаемым дискуссионным проблемам в сочетании с готовностью к конструктивному диалогу и толерантному восприятию иных точек зрения. ПК-7.3. Выделяет концептуальную основу дискуссий в области информационных технологий и робототехники, соотносит с нею свои мировоззренческие установки, гражданскую позицию и социальную мотивацию.	Профессиональный стандарт 01.001 «Педагог (педагогическая деятельность в сфере дошкольного, начального общего, основного общего, среднего общего образования) (воспитатель, учитель)»

3.2. Планируемые результаты обучения по дисциплине, соотнесенные с индикаторами достижения формируемых компетенций

В результате освоения дисциплины обучающийся должен:

Знать:

− основы современных технологий сбора, обработки и представления информации (УК-1 (индикаторы – 1, 2, 3, 4, 5, 6, 7);

− основные понятия информатики, их системно-структурные связи в контексте развития информационно-коммуникационных технологий (ПК-6 (индикаторы – 1, 2, 3);

– теоретические сведения об архитектуре современного компьютера (ПК-6 (индикаторы – 1, 2, 3);

– важнейшие дискуссионные проблемы в научном освоении информатики (трудные вопросы развития и исторического становления научной области «Информатика», прикладные аспекты информатики) (ПК-7 (индикаторы – 1, 2, 3).

Уметь:

– использовать современные информационно-коммуникационные технологии (включая пакеты прикладных программ, локальные и глобальные компьютерные сети) для сбора, обработки и анализа информации (УК-1 (индикаторы – 1, 2, 3, 4, 5, 6, 7);

– использовать программную поддержку курса и оценивать ее методическую целесообразность (УК-1 (индикаторы – 1, 2, 3, 4, 5, 6, 7);

– использовать различного рода прикладные программы и языки программирования для решения учебных задач (ПК-6 (индикаторы – 1, 2, 3);

– анализировать и характеризовать готовый программный продукт в контексте решения учебных задач (ПК-6 (индикаторы – 1, 2, 3);

– выявлять в конкретной решенной задаче ошибки и недочеты, определять их природу (ПК-6 (индикаторы – 1, 2, 3);

– работать с технической и специальной литературой (ПК-7 (индикаторы – 1, 2, 3).

Владеть:

– навыками сопоставления разных источников информации с целью выявления их противоречий и поиска достоверных суждений (УК-1 (индикаторы – 1, 2, 3, 4, 5, 6, 7);

– технологиями алгоритмизации и программирования (ПК-6 (индикаторы – 1, 2, 3);

– спорными и противоречивыми проблемами в области построения научных теорий информатики (ПК-7 (индикаторы – 1, 2, 3);

– механизмами оценки применимости научных положений в прикладных областях (ПК-7 (индикаторы – 1, 2, 3).

4. Объём и содержание дисциплины

Очная форма обучения

Объём дисциплины составляет 2 зачётных единицы (54 час.)

Контактная работа 22,75

Самостоятельная работа 31,5 час.

Практическая подготовка 38,75 час.

Виды учебных занятий и формы промежуточной аттестации	7 сем.	
	Контактная работа с преподавателем	Самостоятельная работа обучающегося
Лекции	10,5	31,25
Практические работы	12	
Зачет	0,25	-
Всего	22,75	31,25

4.1. Содержание дисциплины по разделам (темам), соотнесенное с видами и объемом учебных занятий

Очная форма обучения

№ п/п	Раздел (тема) дисциплины	Семестр	Виды учебных занятий, включая самостоятельную работу обучающихся и объем (в часах)				Формы текущего контроля успеваемости Формы промежуточной аттестации
			Занятия лекционного типа	Занятия семинарского типа		Самостоятельная работа обучающегося	
				Лаб.	Пр.		
1	Основные понятия системного программирования	7	1,5		1,5 ПП	3,5	доклад
2	Общие сведения о программировании микропроцессора, управлении памятью	7	1,5		1,5 ПП	3	доклад
3	Основы программирования	7	3		3	8 ПП	лабораторная

	на языке C++				ПП		работа
4	Средства языка C++ для системного программирования	7	1,5		3 ПП	6,25 ПП	лабораторная работа
5	Библиотека WinAPI как прямой способ взаимодействия приложений пользователя с операционной системой	7	1,5		1,5 ПП	8 ПП	лабораторная работа
6	Многопоточное программирование и механизмы межпроцессорного взаимодействия	7	1,5		1,5 ПП	4,5 ПП	лабораторная работа
ИТОГО:			10,5		12	31,25	экзамен

4.2. Развернутое описание содержания учебного материала по разделам (темам)

Основные понятия системного программирования. Системное программирование. Машинный и байт-код. Регистры, память и логическая адресация. Интерфейс программного обеспечения API. Информация о языках, используемых для системного программирования.

Общие сведения о программировании микропроцессора, управлении памятью. Режимы работы микропроцессора. Реальный режим. Защищенный режим. Инициализация различных режимов и выход из них. Обработка прерываний. Структура и особенности обработчиков прерываний. Системные средства распределения памяти. Организация дочерних процессов. Программы, резидентные в памяти. Организация и взаимодействие резидентных программ. Защита резидентной программы от повторной установки. Выгрузка резидентной программы из памяти.

Основы программирования на языке C++. Синтаксис языка программирования. Структура программы на языке C++. Математические и логические операции C++. Реализация условного оператора. Реализация циклических алгоритмов. Операторы for, while и do..while. Работа со строками. Подпрограммы. Модификаторы доступа. Объекты, классы. Конструкторы и деструкторы. Константность. Конструктор копирования и оператор присваивания.

Средства языка C++ для системного программирования. Механизмы работы с памятью. Как выполняются программы на C++. Стек вызовов. Иллюстрация уязвимости переполнения буфера. Рекурсивные вызовы. Указатели и массивы. Связь массив и указателей. Использование указателей. Ссылки. Динамическая память. Многомерные массивы.

Библиотека WinAPI как прямой способ взаимодействия приложений пользователя с операционной системой. Структуры и функции WinAPI. Работа с WinAPI из C++. Использование документации MSDN. Сообщения окна: WM_CREATE, WM_DESTROY, WM_CLOSE, WM_SIZE, WM_COMMAND. Подкласс окна и элементов управления. Методы для работы с окнами. Службы Windows. Функции работы с содержимым реестра. Сообщения об ошибках и их обработка.

Многопоточное программирование и механизмы межпроцессорного взаимодействия. Параллельное программирование. Запуск потока и присоединение к потоку.

Совместный доступ потоков к данным. Двоичные семафоры. Предотвращение тупиков. Потокобезопасный стек. Функции WinAPI для реализации синхронизации потоков. Асинхронные задачи в языке C++.

5. Образовательные технологии, в том числе перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине

В соответствии с требованиями ФГОС ВО в программе данного курса рекомендуется использовать следующие образовательные технологии: лекции, практические занятия, самостоятельная работа студентов.

В преподавании дисциплины существуют темы, изучение которых должно обязательно начинаться с теоретической подготовки. Чтобы повысить результативность изучения теоретического материала, применяются следующие приемы:

- лекции сопровождаются компьютерными презентациями;
- применяются методики «мозгового штурма», разбора практических ситуаций;
- используются краткие сообщения студентов по новейшим разработкам (по материалам Интернет) с использованием демонстрационных материалов.

Основной и самой результативной формой обучения являются практические занятия. Для преподавания дисциплины создан учебно-методический комплекс, обеспечивающий предоставление информации (электронные учебники и пособия) и управление деятельностью обучаемого.

Разработанные практические задания включают в себя обучающие тексты, набор пошаговых инструкций, учебных задач и заданий, демонстрационный материал.

Такой подход к организации проведения практических занятий стимулирует и организует самостоятельную работу студента. Достигаются не только учебные, но и развивающие цели, т.к. формируется умения по самоорганизации учебной деятельности.

Перечень информационных технологий, используемых при осуществлении образовательного процесса по дисциплине: мультимедиа технологии; технологии смешанного обучения.

6. Учебно-методическое обеспечение самостоятельной работы обучающихся

Самостоятельная работа обучающихся по дисциплине осуществляется в следующих формах:

– Конспектирование и реферирование литературных источников по тематике учебного курса;
– Установка и настройка среды разработки для системного программирования;
– Решение задач, связанных с написанием программ на языке C++;
– Решение задач, связанных с написанием программ, работающих с системным программным обеспечением;
– Изучение и анализ предложенных электронных образовательных ресурсов, содержащих информацию по системному программированию;
– Повторение материала с использованием конспектов.

Полностью весь методический материал по обеспечению самостоятельной работы студентов приводится в приложении 1 к настоящей программе. Доступ к методическим материалам осуществляется через сайт Шуйского филиала ИвГУ, в том числе через ЭБС Шуйского филиала ИвГУ на платформе «Руконт».

7. Характеристика оценочных средств для текущего контроля успеваемости и промежуточной аттестации по дисциплине

Текущая аттестация осуществляется с использованием накопительного рейтингового оценивания. Критерии оценки деятельности обучающихся разрабатываются в зависимости от уровня их информационно технологической и дидактической подготовки обучающихся.

Для текущего контроля успеваемости применяются:

- Практические работы на компьютере: 10 лабораторных работ по 6 баллов (60 баллов);
- Доклад по предложенной теме (10 баллов);

Помимо этого учитывается:

- степень активности и самостоятельности обучающегося на занятиях – 15 баллов,
- посещаемость (отсутствие пропусков занятий) – 15 баллов.

Промежуточная аттестация по дисциплине (зачет) проводится с использованием рейтинговой шкалы оценки. На зачете по дисциплине «Системное программирование» контролируемые знания, умения и владения, характеризующие этапы формирования компетенций в процессе освоения образовательной программы. Зачет проводится в форме устного собеседования по темам (10 баллов), представленным в приложении, а также выполнения практических заданий (20 баллов).

Общая накопительная оценка складывается из баллов текущей успеваемости в семестре по учебной дисциплине и баллов, полученных студентами на зачете, таким образом, чтобы результирующий балл не превышал значения 100.

Шкала оценки результатов по дисциплине «Системное программирование» при рейтинговой системе оценивания

Повышенный уровень	Базовый уровень	Пороговый уровень	Ниже порогового уровня
зачтено	зачтено	зачтено	не зачтено
85-100 баллов	71-84 балла	61-70 баллов	менее 61 балла

8. Учебно-методическое и информационное обеспечение дисциплины

Учебная и учебно-методическая литература:

- ***основная литература:***

1. Зюзьков, В. М. Программирование : учебное пособие / В.М. Зюзьков ; Министерство образования и науки Российской Федерации, Томский Государственный Университет Систем Управления и Радиоэлектроники (ТУСУР). - Томск : Эль Контент, 2013. - 186 с. : ил. - Библиогр. в кн. - ISBN 978-5-4332-0141-5 ; То же [Электронный ресурс]. - URL: http://biblioclub.ru/index.php?page=book&id=480616

2. Калентьев, А. А. Новые технологии в программировании : учебное пособие / А. А. Калентьев, Д. В. Гарайс, А. Е. Горяинов ; Министерство образования и науки Российской Федерации, Томский Государственный Университет Систем Управления и Радиоэлектроники (ТУСУР). - Томск : Эль Контент, 2014. - 176 с. : схем., ил. - Библиогр.: с. 166-169. - ISBN 978-5-4332-0185-9 ; То же [Электронный ресурс]. - URL: http://biblioclub.ru/index.php?page=book&id=480503

3. Сергиевский, Г. М. Функциональное и логическое программирование : учебное пособие для студентов вузов / Г. М. Сергиевский, Н. Г. Волченков. - Москва : Академия, 2010. - 320 с.

- ***дополнительная литература:***

1. Голицына О. Л. Основы алгоритмизации и программирования : учебное пособие / О. Л. Голицына, И. И. Попов. - Москва : Форум : Инфра-М, 2005. - 432 с.

2. Ефимова, Е.А. Основы программирования на языке Visual Prolog / Е.А. Ефимова. - 2-е изд., испр. - Москва : Национальный Открытый Университет «ИНТУИТ», 2016. - 266 с. : ил. - Библиогр. в кн. ; То же [Электронный ресурс]. - URL: http://biblioclub.ru/index.php?page=book&id=428996
3. Зыков, С.В. Введение в теорию программирования. Объектно-ориентированный подход / С.В. Зыков. - 2-е изд., испр. - Москва : Национальный Открытый Университет «ИНТУИТ», 2016. - 189 с. : схем. - (Основы информационных технологий). - Библиогр. в кн. - ISBN 5-9556-0009-4 ; То же [Электронный ресурс]. - URL: http://biblioclub.ru/index.php?page=book&id=429073
4. Зыков, С.В. Введение в теорию программирования. Функциональный подход / С.В. Зыков. - 2-е изд., испр. - Москва : Национальный Открытый Университет «ИНТУИТ», 2016. - 153 с. : ил. - (Основы информационных технологий). - Библиогр. в кн. - ISBN 5-9556-0009-4 ; То же [Электронный ресурс]. - URL: http://biblioclub.ru/index.php?page=book&id=429119
5. Иванова, Г. С. Объектно-ориентированное программирование : учебник для вузов / Г. С. Иванова, Т. Н. Ничушкина, Е. К. Пугачев ; под ред. Г. С. Ивановой. - 2-е изд. ; перераб. и доп. - Москва : Изд-во МГТУ им. Н. Э. Баумана, 2003. - 368 с.
6. Иванова, Г. С. Технология программирования : учебник / Г. С. Иванова. - Москва : КНОРУС, 2011. - 336 с.
7. Мейер, Б. Объектно-ориентированное программирование и программная инженерия / Б. Мейер. - 2-е изд., испр. - Москва : Национальный Открытый Университет «ИНТУИТ», 2016. - 286 с. : ил. ; То же [Электронный ресурс]. - URL: http://biblioclub.ru/index.php?page=book&id=429034
8. Рублев, В.С. Языки логического программирования : учебное пособие / В.С. Рублев. - Москва : Интернет-Университет Информационных Технологий, 2008. - 115 с. ; То же [Электронный ресурс]. - URL: http://biblioclub.ru/index.php?page=book&id=234653
9. Слабнов, В.Д. Программирование на С++ : лекции / В.Д. Слабнов ; Институт экономики, управления и права (г. Казань). – Казань : Познание, 2012. – 136 с. : табл., схем. – Режим доступа: по подписке. – URL: http://biblioclub.ru/index.php?page=book&id=364222

Интернет-ресурсы:

1. М. В. Свиркин, А. С. Чуркин. Программирование под Windows в среде Visual C++ 2005 / М. В. Свиркин, А. С. Чуркин. – НОУ «ИНТУИТ». Режим доступа: https://www.intuit.ru/studies/courses/495/351/info
2. Единая коллекция цифровых образовательных ресурсов http://school-collection.edu.ru/
3. Единое окно доступа к образовательным ресурсам. http://window.edu.ru/
4. Образовательные сайты и порталы. http://school2100.com/
5. Сайты учителей. http://rus.1september.ru/
6. Профессиональный сайт учителей «Методисты». http://nsportal.ru/
7. Социальная сеть работников образования. http://nsportal.ru/shkola

Современные профессиональные базы данных и информационно-справочные системы:

ЭБС «Университетская библиотека онлайн»;

Коллекция Шуйского филиала ИвГУ на национальном цифровом ресурсе «Руконт».

Программное обеспечение:

В процессе изучения дисциплины используются офисные пакеты Microsoft Office, среда программирования wxDev-C++, программа для просмотра и чтения файлов PDF Adobe

Acrobat Reader DC, программа для воспроизведения флэш-анимации в браузерах Adobe Flash Player, веб-обозреватели Google Chrome, Mozilla Firefox, Антивирус Касперского, 1С Университет.

9. Материально-техническое обеспечение дисциплины

1. Учебная аудитория для проведения занятий лекционного типа, укомплектованная специализированной мебелью и техническими средствами обучения, служащими для предоставления учебной информации большой аудитории.
2. Учебная аудитория для проведения занятий семинарского типа, укомплектованная специализированной мебелью и техническими средствами обучения.
3. Учебная аудитория для проведения текущего контроля и промежуточной аттестации, укомплектованная специализированной мебелью и техническими средствами обучения.
4. Помещение для самостоятельной работы, оснащенное специализированной мебелью, компьютерной техникой с возможностью подключения к сети «Интернет» и обеспечением доступа в электронную информационно-образовательную среду организации.
5. Учебно-наглядные пособия для занятий лекционного типа, обеспечивающие тематические иллюстрации: презентации по языку программирования С++ и основам системного программирования.

Автор рабочей программы дисциплины: А.В. Ермошин, доцент кафедры математики, информатики и методики обучения, кандидат технических наук, доцент.

Программа рассмотрена на заседании кафедры математики, информатики и методики обучения «16» февраля2023г., протокол № 7

Программа обновлена
протокол заседания кафедры № _____ от «_____» _____20 __ г.
Согласовано:
Руководитель ОП _____ С.В. Белов
(подпись)

Программа обновлена
протокол заседания кафедры № _____ от «_____» _____20 __ г.
Согласовано:
Руководитель ОП _____ С.В. Белов
(подпись)

Программа обновлена
протокол заседания кафедры № _____ от «_____» _____20 __ г.
Согласовано:
Руководитель ОП _____ С.В. Белов
(подпись)

Приложение 1

Методические указания для обучающихся по освоению дисциплины

На самостоятельную работу студентов по курсу «Системное программирование» отводится более 60% учебного времени от общей трудоемкости курса. Сопровождение самостоятельной работы студентов по данному курсу может быть организовано в следующих формах:

– согласование индивидуальных заданий (виды и темы заданий, сроки представления результатов) самостоятельной работы студента в пределах часов, отведенных на самостоятельную работу;

– консультации (индивидуальные и групповые), в том числе с применением дистанционных технологий взаимодействия;

– промежуточный контроль хода выполнения заданий, строящихся на основе различных способов самостоятельной информационной деятельности в открытой информационной среде.

Преподаватель определяет содержание самостоятельной работы, график её выполнения; создает сетевую информационную и коммуникационную среду для организации самостоятельной работы.

В процессе организации самостоятельной работы особое внимание уделяется не только развитию информационной культуры, но и формированию готовности к кооперации, к работе в коллективе в сетевом пространстве, для чего целесообразно стимулировать студентов к совместной деятельности в малых группах.

Одной из важнейших особенностей заданий для самостоятельной работы по курсу «Системное программирование» является их ориентация на формирование способности оценивать приоритеты и ограничения при выборе средств информационных технологий для решения управленческих задач, а также на мотивирование самообразовательной деятельности с использованием ИКТ.

К типовым заданиям для самостоятельной работы студентов относятся:

– Конспектирование и реферирование литературных источников по тематике учебного курса;
– Установка и настройка среды разработки для системного программирования;
– Решение задач, связанных с написанием программ на языке C++;
– Решение задач, связанных с написанием программ, работающих с системным программным обеспечением;
– Изучение и анализ предложенных электронных образовательных ресурсов, содержащих информацию по системному программированию;
– Повторение материала с использованием конспектов.

Перечень вопросов для самостоятельного изучения

1. Особенности языка C++
2. Структура программы в языке C++
3. Переменные в C++: понятие, объявление, инициализация, основные типы
4. Арифметические операции в C++
5. Операторы ветвления if и множественного ветвления switch
6. Операторы цикла for, while, do..while
7. Указатели и ссылки в C++
8. Функции: понятие, описание, вызов, передача параметров
9. Понятие стека. Стек функций

10. Массивы в C++, доступ к элементу массива по адресу и индексу
11. Классы и объекты в C++. Открытые и закрытые переменные
12. Стандартные структуры данных в C++
13. Особенности работы с многомерными массивами
14. Библиотеки для низкоуровневого программирования в C++
15. WinAPI: понятие, история возникновения и развития
16. Wine как реализация WinAPI в UNIX-подобных системах
17. Функции WinAPI: WinMain, GetMessage, DispatchMessage
18. Сообщения и их обработка в WinAPI: WM_CREATE, WM_DESTROY, WM_CLOSE, WM_SIZE, WM_COMMAND
19. Формирование сообщений, примеры. Очереди. Цикл сообщений, основные компоненты
20. Окна, классы и типы окон. Регистрация класса окна. Основные элементы окна
21. Понятие о системном реестре. Функции для работы с системным реестром Windows
22. Параллельное программирование
23. Потоки: понятие, принципы работы и управления

Задания для организации практических занятий:

Практическая работа 1: «Основы программирования на языке C++»

1. Напишите функцию power, реализующую возведение целого числа в неотрицательную целую степень. Функция power должна принимать на вход два целых числа и возвращать целое число. При выполнении задания учтите, что функция обязательно должна называться power, функция ничего не должна читать со входа или выводить. Вы можете определять вспомогательные функции, если они вам нужны. Реализовывать функции main не нужно. Библиотеку cmath (и math.h) использовать запрещено.

2. Напишите программу, которая суммирует целые числа. На вход программе подаются целые числа в следующем формате: на первой строке идет целое число T — количество тестов, далее следует T строк, в каждой из которых через пробел идут два целых числа a_i и b_i. На выводе для каждой из T строк нужно вывести сумму a_i+b_i в том порядке, в котором пары поступают на вход. Ничего, кроме этого, выводить не нужно.

3. Напишите программу для решения квадратных уравнений вида $ax^2+bx+c=0$ (относительно x). На вход программа получает три целых числа: a, b и c, соответственно. При этом гарантируется, что $a\neq0$. На вывод программа должна вывести два вещественных корня уравнения, разделённые пробелом. Если вещественных корней нет, то программа должна вывести строку "No real roots". Если у уравнения имеется только один корень (кратный корень), то программа должна вывести его дважды. Порядок вывода корней не важен. Ничего, кроме этого, выводить не нужно. Для вычислений с плавающей точкой используйте тип double. При выполнении задания вам может оказаться полезной функция sqrt из заголовочного файла cmath.

4. Напишите программу, которая посимвольно читает из std::cin, пока не достигнет конца потока ввода, и заменяет несколько подряд идущих пробелов одним и выводит полученный результат в std::cout. Никаких других символов, кроме пробелов удалять не нужно. При выполнении задания вам не разрешается пользоваться дополнительной памятью, а именно:

массивами, стандартными контейнерами и строками, даже если вы уже с ними знакомы. Вы можете определять любые вспомогательные функции, если они вам нужны.
5. Напишите программу для вычисления целочисленного логарифма по основанию 2. На вход программе в первой строке подается число T, далее следует T строк с тестами. Каждый тест состоит из одного целого положительного числа $a_i < 10^9$, для каждого a_i нужно вывести на отдельной строке такое наибольшее число p, что $2^p \leq a_i$. Гарантируется, что $a_i \geq 1$. При решении задачи вы можете определять любые вспомогательные функции, если они вам нужны, более того рекомендуется вынести вычисление логарифма в отдельную функцию. Библиотеку cmath (и math.h) использовать запрещено.

6. Напишите программу, которая ищет наибольший общий делитель (НОД) с использованием алгоритма Евклида. На вход программе подаются целые числа в следующем формате: на первой строке идет целое число T — количество тестов, далее следует T строк, в каждой из которых через пробел идут два целых числа a_i и b_i. На выводе для каждой из T строк нужно вывести наибольший общий делитель данных чисел в том порядке, в котором пары поступают на вход.

Критерии оценки:
За каждое верно и самостоятельно выполненное задание студент получает 1 балл.
Всего баллов – 6 (Максимально)

Практическая работа 2: «Работа с массивами на языке C++»

1. Пользователь вводит с клавиатуры массив из 20 действительных чисел. Отсортировать данный массив и подсчитать количество уникальных чисел в массиве.

2. Пользователь многократно вводит число (количество очков, набранных футбольной командой в чемпионате) и строку (название футбольной команды высшей лиги России). Известно, что нет команд с равным числом очков. Выведите на экран список призёров.

3. Пользователь вводит с клавиатуры массив из 30 действительных чисел. Отсортировать с помощью простого выбора элементы массива, стоящие на нечетных местах.

4. Дана квадратная вещественная матрица размерности n. Найти количество нулевых элементов, стоящих: выше главной диагонали; ниже главной диагонали; выше и ниже побочной.

5. Дан двумерный массив A[1..N,1..N]. Составить программу, которая прибавила бы каждому элементу данной строки элемент, принадлежащий этой строке и главной диагонали.

6. Дана матрица NxM. Переставляя ее строки и столбцы, переместить наибольший элемент в верхний левый угол. Определить можно ли таким же образом поместить минимальный элемент в нижний правый угол.

Критерии оценки:
За каждое верно и самостоятельно выполненное задание студент получает 1 балл.
Всего баллов – 6 (Максимально)

Практическая работа 3: «Структуры данных на языке С++»

1. Описать структуру с именем NOTE, содержащую следующие поля:
- NAME — фамилия, имя;
- TELE — номер телефона;
- BDAY — день рождения (массив из трех чисел).

Написать программу, выполняющую следующие действия:
- ввод с клавиатуры данных в массив BLOCKNOTE, состоящий из восьми элементов типа NOTE; записи должны быть упорядочены по трем первым цифрам номера телефона;
- вывод на экран информации о человеке, чья фамилия введена с клавиатуры;
- если такого нет, выдать на дисплей соответствующее сообщение.

2. Информация по N заводам города задается строкой следующего вида: фамилия, средний возраст, специальность, средний оклад. Ввести информацию по заводам, посчитать количество слесарей и токарей. Напечатать эти значения

3. Описать, используя структуру данных запись, данные на учеников (фамилия, улица, дом, квартира). Составить программу, определяющую сколько учеников живет на улице Свердлова, списки учеников, живущих в доме номер 45.

4. Хранятся сведения о лесе: вид дерева, общая численность, численность здоровых деревьев. Составить программу вычисления: 1) суммарного числа деревьев на контрольном участке; 2) суммарного числа здоровых деревьев; 3) относительную численность (%) больных деревьев; 4) относительную численность (%) различных видов, в том числе больных (%) для каждого вида.

Критерии оценки:
За каждое верно и самостоятельно выполненное задание студент получает 1,5 балла.
Всего баллов – 6 (Максимально)

Практическая работа 4: «Алгоритмические структуры и работа с файлами в языке С++»

1. Подсчитать количество элементов в стеке.

2. Сформировать стек, содержащий строки и сохранить его в текстовом файле. Восстановить стек, содержащий строки, из текстового файла.

3. В данном тексте проверить соответствие открытия и закрытия скобок.

4. Сформировать два списка, отсортировать их объединить в один, не нарушая порядка.

5. Написать функцию, которая проверяет, упорядочены ли элементы списка по алфавиту.

6. Организовать структуру «Бинарное дерево» с использованием указателей. Написать функцию, которая находит наибольший элемент дерева.

Критерии оценки:
За каждое верно и самостоятельно выполненное задание студент получает 1 балл.
Всего баллов – 6 (Максимально)

Практическая работа 5: «Объектно-ориентированное программирования на языке C++»

1. Построить систему классов для описания плоских геометрических фигур: круга, квадрата, прямоугольника. Предусмотреть методы для создания объектов, перемещения на плоскости и изменения размеров.

2. Построить описание класса, содержащего информацию о почтовом адресе организации. Предусмотреть возможность раздельного изменения составных частей адреса, создания и уничтожения объектов этого класса.

3. Составить описание класса для объектов-векторов, задаваемых координатами концов в трехмерном пространстве. Обеспечить операции сложения и вычитания векторов с получением нового вектора (суммы или разности), вычисления скалярного произведения двух векторов, длины вектора, косинуса угла между векторами.

4. Составить описание класса прямоугольников со сторонами, параллельными осям координат. Предусмотреть возможность перемещения прямоугольников на плоскости, изменения размеров, построения наименьшего прямоугольника, содержащего два заданных прямоугольника, и прямоугольника, являющегося общей частью (пересечением) двух прямоугольников.

Критерии оценки:
За каждое верно и самостоятельно выполненное задание студент получает 1,5 балла.
Всего баллов – 6 (Максимально)

Практическая работа 6: «Получение данных о системе из C++ с помощью WinAPI»

1. Вывести количество времени, прошедшее с момента запуска операционной системы (формат: сутки, часы, минуты, секунды)

2. Написать программу, которая выводит текущее время в MessageBox.

3. Вывести на экран имя компьютера и текущего пользователя

Критерии оценки:
За каждое верно и самостоятельно выполненное задание студент получает 2 балла.
Всего баллов – 6 (Максимально)

Практическая работа 7: «Работа с объектами ядра из C++ с помощью WinAPI»

1. Создание одного из объектов ядра и получение описателя(handle) на него (функции Create: - CreateProcess, CreateMutex и другие)

2. Проверка, что объект создан, то есть описатель не равен NULL, иначе получить код ошибки (функция GetLastError), форматировать сообщение об ошибке (FormatMessage) и вывести его (с помощью MessageBox или в консоль). Получение параметров описателя объекта и вывод их на экран
3. Дублирование описателя этого же объекта (функция DuplicateHandle)
4. Проверка равны ли первоначальный описатель и его клон, вывод на экран результата
5. Закрытие первого описателя (CloseHandle). Проверка существует ли еще объект ядра, то есть вызов любой функции, связанной с этим объектом (например для объекта File - ReadFile), проверка результата выполнения функции и вывод на экран сообщения об удачном/неудачном выполнении
6. Закрытие второго описателя. Проверка существует ли еще объект ядра и вывести результат на экран (аналогично предыдущему пункту)

Критерии оценки:
За каждое верно и самостоятельно выполненное задание студент получает 1 балл.
Всего баллов – 6 (Максимально)

Практическая работа 8: «Многопоточное программирование в C++ с помощью WinAPI»

1. Создать поток, который выводит два сообщения через MessageBox с интервалом в 5 секунд. После окончания его работы основной поток процесса выводит сообщение об этом.
2. Создать поток и вывести в нем параметры текущего процесса: заголовок окна, расстояния по осям X и Y от левого верхнего угла, ширину и высоту окна, дескрипторы стандартных устройств ввода и вывода.
3. Вывести параметры текущего процесса: заголовок окна, расстояния по осям X и Y от левого верхнего угла, ширину и высоту окна, дескрипторы стандартных устройств ввода и вывода. Затем создать процесс с заданными параметрами заголовка окна, ширины, высоты и смещения от левого верхнего угла экрана. После чего вывести те же параметры, что и для первого процесса.

Критерии оценки:
За каждое верно и самостоятельно выполненное задание студент получает 2 балла.
Всего баллов – 6 (Максимально)

Практическая работа 9: «Многопоточное программирование в C++ с помощью WinAPI»

1. Создать две функции потоков, которые выполняют следующие расчеты:

Поток 1: a=(b+3); b = (b-1);
Поток 2: b = (b+2);

Каждая функция потока вычисляет свою формулу в цикле из ста итераций. При вычислении значения переменных a и b выводить на экран.

2. Произвести те же вычисления при помощи критических секций. После этого вывести значения переменных a и b на экран.

3. Написать две программы:

Программа 1 создает объект ядра «событие» выводит на экран текущее время и делает задержку на пять секунд, после чего освобождает «событие»; Перед задержкой запустить Программу 2;

Программа 2 ждет, пока освободится объект ядра «событие», после чего выводит на экран текущее время.

4. Создать 3 функции потоков, каждая из которых выводит в цикле из 25 итераций сообщение «Это поток № номер_потока». Сообщение заносить в строку, на которую указывает глобальная переменная str. Сделать синхронизацию потоков при помощи мьютексов.

Критерии оценки:
За каждое верно и самостоятельно выполненное задание студент получает 1,5 балла.
Всего баллов – 6 (Максимально)

Практическая работа 10: «Работа с процессами из C++ с помощью WinAPI»

1. Создать два дочерних процесса и соединить их анонимным каналом. Для этого использовать переопределение стандартных устройств ввода/вывода. Основная программа создает канал, процессы и передает им описатели чтения/записи канала. Вторая программа получает от родительской описатель записи в канал в качестве стандартного устройства вывода и передает по каналу данные из файла (определяется программистом). Третья программа получает описатель чтения из канала в качестве стандартного устройства ввода, считывает через него информацию из канала и выводит ее на экран.
2. Вести широковещательную рассылку сообщений по сети при помощи почтовых ящиков. Сервер создает почтовый ящик и периодически записывает туда сообщения. Клиенты подключаются к этому ящику, считывают полученные сообщения и выводят их на экран.
3. Реализовать механизм клиент/сервер при помощи именованных каналов. Сервер передает клиенту аргументы, при помощи которых клиент проводит определенные вычисления (например, y=3*a+5*b-2*c). Результат вычислений передается обратно по каналу серверу, который осуществляет вывод на экран.

Критерии оценки:
За каждое верно и самостоятельно выполненное задание студент получает 2 балла.
Всего баллов – 6 (Максимально)

Примерная тематика докладов:

1. История языка программирования C++
2. Обзор среды программирования Microsoft Visual Studio для C++
3. Обзор среды программирования NetBeans
4. Обзор среды программирования Qt Creator
5. Обзор среды программирования Code::Blocks

6. Обзор среды программирования Code::Blocks
7. Обзор среды программирования Dev-C++ и её расширенной версии wxDev-C++
8. Прерывание и его природа
9. Характеристики реального режима работы микропроцессора
10. Системные средства распределения памяти
11. Физическая и логическая организация адресного пространства
12. Основы организации резидентных программ. Проблемы разработчиков резидентных программ и обработчиков прерываний
13. Расширения традиционной архитектуры Intel
14. Задачи по управлению процессами и организация дочерних процессов
15. API как средство интеграции приложений
16. API операционных систем. Проблемы, связанные с многообразием API
17. Использование API в web-разработке
18. Двоичный (бинарный) интерфейс приложений

Доклад

Задание:

Подготовить доклад по предложенной тематике.

Критерии оценки

Оценки в баллах	Критерии
2	Самостоятельность в подготовке материала
3	Аргументированность и полнота материала
2	Использование современной и актуальной литературы
3	Качество презентации
10	**Итого**