

НЕГОСУДАРСТВЕННОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ
ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
«САНКТ-ПЕТЕРБУРГСКИЙ ГУМАНИТАРНЫЙ
УНИВЕРСИТЕТ ПРОФСОЮЗОВ»

Кафедра Информатики и математики

РАБОЧАЯ ПРОГРАММА ДИСЦИПЛИНЫ
Информатика и программирование

Основная профессиональная образовательная программа
высшего образования программы бакалавриата
по направлению

09.03.03 «Прикладная информатика»

Профиль подготовки «Прикладная информатика в экономике»

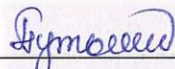
Квалификация:
Бакалавр

Согласовано:
Руководитель ОПОП по направлению
09.03.03 – «Прикладная информатика»
Профиль «Прикладная информатика
в экономике»

 /Путькина Л.В.

Рассмотрена и утверждена на заседании кафедры

«01» июня 2020 г., протокол № 10

Зав. кафедрой  /Путькина Л.В.

Рекомендована решением
Методического совета

«15» июня 2020 г., протокол № 10

Секретарь МС  Волкова А.М.

Авторы-разработчики:

 /Мокрый В.Ю.

Санкт-Петербург

СТРУКТУРА

1. Цель и задачи освоения дисциплины
2. Место дисциплины в структуре ОПОП
3. Требования к результатам освоения дисциплины
4. Тематический план изучения дисциплины
5. Содержание разделов и тем дисциплины
6. План подгрупповых (лабораторных) занятий
7. Образовательные технологии
8. План самостоятельной работы студентов
9. Контроль знаний по дисциплине
10. Учебно-методическое и информационное обеспечение дисциплины
11. Материально-техническое обеспечение дисциплины

Учебно-методическое обеспечение самостоятельной работы студентов

1. Методические рекомендации по организации самостоятельной работы студентов
2. Методические рекомендации по подготовке к практическим (подгрупповым) занятиям
3. Методические рекомендации по написанию контрольных работ
4. Методические рекомендации по написанию курсовой работы

Оценочные и методические материалы

1. Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы
2. Описание показателей и критериев оценивания компетенций, шкал оценивания
3. Типовые контрольные задания и методические материалы, процедуры оценивания знаний, умений и навыков

Глоссарий

Методические рекомендации для преподавателя по дисциплине

1. Цель и задачи освоения дисциплины:

Целью изучения дисциплины «Информатика и программирование» является расширение мировоззрения и формирование у студентов самостоятельного мышления в области информационных подходов к анализу процессов и явлений профессиональной сферы и информационных технологий.

Основные задачи дисциплины – получение систематических знаний о информационных процессах, системах, средствах и технологиях; приобретение специальных знаний и умений, необходимых для работы в фирмах и организациях профессионально работающих в информационной сфере или использующих новые информационные технологии.

2. Место дисциплины в структуре ОПОП

Междисциплинарные связи с обеспечиваемыми (последующими) дисциплинами

№ п/п	Наименование обеспечиваемых (последующих) дисциплин	№ № разделов данной дисциплины, необходимых для изучения обеспечиваемых дисциплин			
		1	2	3	4
1.	Информационные системы и технологии	+	+	+	+
2.	Программная инженерия		+	+	+
3.	Структуры данных в предметной области	+	+	+	+
4.	Объектно-ориентированное программирование	+		+	+

3. Требования к результатам освоения дисциплины

Процесс изучения дисциплины направлен на формирование следующих компетенций с установленными к ним индикаторами:

Компетенции и индикаторы их достижения

Категория компетенций	Код и наименование компетенции	Код и наименование индикатора достижения 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. Знает основные положения информатики как научной дисциплины, изучающей процессы обработки информации, принципы организации и функционирование современных средств обработки информации, базовые принципы информационного анализа процессов и явлений предметной области.
		УК -1.2. Умеет соотносить разнородные явления и систематизировать их в рамках избранных видов профессиональной деятельности.
		УК -1.3. Имеет практический опыт работы с информационными источниками, опыт научного поиска,

		создания научных текстов.
Исследовательская деятельность	ОПК-1. Способен применять естественнонаучные и общеинженерные знания, методы математического анализа и моделирования, теоретического и экспериментального исследования в профессиональной	ОПК-1.1. Знает основные положения информатики как научной дисциплины, изучающей процессы обработки информации, принципы организации и функционирование современных средств обработки информации, базовые принципы информационного анализа процессов и явлений предметной области; методы проектирования и разработки алгоритмов и программ
		ОПК-1.2. Умеет решать задачи обработки данных; составлять алгоритмы, проектировать, программировать программы на одном из языков высокого уровня с использованием современных технологий программирования
		ОПК-1.3. Владеет навыками информационного анализа процессов и явлений, составления алгоритмов.
Исследовательская деятельность	ОПК-4. Способен участвовать в разработке стандартов, норм и правил, а также технической документации, связанной с профессиональной деятельностью	ОПК-4.1. Знает основные положения информатики как научной дисциплины, изучающей процессы обработки информации, принципы организации и функционирование современных средств обработки информации, базовые принципы информационного анализа процессов и явлений предметной области; методы проектирования и разработки алгоритмов и программ
		ОПК-4.2. Умеет решать задачи обработки данных с помощью современных инструментальных средств конечного пользователя; составлять алгоритмы, проектировать, программировать и тестировать программы на одном из языков высокого уровня с использованием современных технологий программирования
		ОПК-4.3. Владеет навыками документирования программ на одном из языков высокого уровня с использованием современных программных средств
Исследовательская деятельность	ОПК-7. Способен разрабатывать алгоритмы и программы, пригодные для практического применения	ОПК-7.1. Знает принципы организации проектирования и содержание этапов процесса разработки программных комплексов дисциплины, задачи и методы исследования и обеспечения

		качества и надежности программных компонентов, основы разработки программных комплексов.
		ОПК-7.2. Умеет проводить анализ и проектирование современного программного обеспечения, определять его качественные показатели, организовывать процесс разработки и вести документацию в соответствии с современными стандартами.
		ОПК-7.3. Владеет навыками программирования, отладки и тестирования программного обеспечения.

4. Тематический план изучения дисциплины

См. приложение

5. Содержание разделов и тем дисциплины

РАЗДЕЛ 1 (Модуль 1). Информатика и информационные технологии

Тема 1. Информация и информационные процессы

Информатика как фундаментальная наука и область практической деятельности. Общекультурная роль информатики. Универсальность подходов и методов информатики. Информационные технологии как форма реализации информационных подходов. Понятие об информационном обществе. Роль и место информатики и информационных технологий в профессиональной деятельности. Занятость в информационной сфере. Экономика, этика и правовые нормы информационной деятельности.

Определения понятия информация. Информационные системы и технологии. Этапы развития информационных технологий. Фазы информационного процесса для получения информации. Этапы преобразования информации. Данные и знания. Синтаксический аспект информации. Определение количества информации по Шеннону. Формулы кодирования и Хартли, примеры их применения. Семантический и прагматический аспекты информации.

Представление дискретной информации в компьютере. Принципы сжатия. Примеры представления текстовой и графической информации. Представление непрерывной информации в компьютере. Емкость канала передачи. Примеры представления звуковой информации.

Информационный подход к анализу процессов и явлений. Основные стадии анализа. Пример информационного анализа. Качественные и количественные факторы. Оцифровка и нормализация значений факторов. Правила для выбора шкал оцифровки. Коэффициенты значимости факторов, способы вычисления оценок.

Тема 2. Программные средства обработки информации

Состав и функции программного обеспечения компьютера. Понятие операционной среды. Классификация операционных систем. Основные группы прикладных программ. Совместимость прикладных программ с операционными системами. Файловые системы. Альтернативность процедур выполнения операций в операционной среде.

Унифицированные функции интерфейса прикладных программ. Способы организации обмена данными между прикладными программами. Связь и внедрение объектов. Настройка операционной системы. Краткая характеристика функций сервисных программ. Файловые менеджеры. Командные файлы и макросредства. Способы защиты информации при ее хранении и обработке на ПК. Восстановление информации. Защита информации от несанкционированного доступа. Резервирование информации. Методы

сжатия информации. Архивация программ и данных. Способы защиты данных от компьютерных вирусов. Методика использования антивирусных средств.

Понятие электронного документа. Системы обработки документов. Работа с текстовым редактором. Основные операции по редактированию текстовых документов. Шрифтовое оформление текста. Стилиевое оформление. Таблицы. Структура сложного документа. Дополнительные возможности.

Автоматизация расчетов над табличными данными. Автоматизация простых математических расчетов. Электронные таблицы. Понятия клетки, адреса, формулы. Применение математических, статистических и экономических функций. Работа с электронной таблицей как с базой данных. Представление табличных данных в виде диаграмм. Дополнительные возможности.

РАЗДЕЛ 2 (Модуль 2). Алгоритмизация и программирование

Тема 3. Постановка и алгоритмизация прикладных задач

Определения понятия задача. Фазы постановки задач. Основные стадии компьютерного решения прикладной задачи. Общее понятие алгоритма. Свойства компьютерных алгоритмов. Теория алгоритмов. Машина Тьюринга. Теория вычислительной сложности. Методика исследования вычислительной сложности задачи.

Способы представления алгоритмов. Блок-схемы. Основные алгоритмические конструкции: следование, цикл, переход, ветвление. Стандартные алгоритмы. Язык программирования как средство представления алгоритма. Классификация языков программирования. Подходы к выбору языка. Версии языка. Содержание и порядок выполнения отдельных этапов создания программы. Компоненты и возможности системы программирования.

Тема 4. Методы и языки программирования

Эволюция технологий программирования. Стадии преобразования программных модулей. Понятие системы программирования. Процесс трансляции программ. Компиляторы и интерпретаторы. Классификация языков программирования. Основные компоненты языка программирования: алфавит, словарь, набор операций, выражения, операторы, система синтаксических правил. Семантика программы.

Основные элементы программы: описания данных, описания процедур, цепочки операторов, тексты процедур, внутренние комментарии. Последовательность размещения элементов программы. Использование библиотек готовых программных модулей. Классификация обрабатываемых данных по типу. Способы описания типов. Понятие стандартных типов языка программирования и типов пользователя.

Программирование основных алгоритмических конструкций. Структура оператора присваивания. Присваивание значений переменным. Понятие совместимости типов. Программирование ввода/вывода. Операторы перехода и ветвления. Структура и синтаксис полного и неполного условного оператора. Понятие заголовка и тела цикла. Варианты и синтаксис операторов цикла. Применение, структура и синтаксис и вложенных циклов. Описание одномерных и двумерных массивов. Размещение массивов в памяти. Работа с одномерными и двумерными массивами в циклах. Ввод/вывод структурированных данных. Назначение процедур. Правила описания процедур и обращения к ним.

Классификация ошибок в программе. Подходы к формальному доказательству правильности программ. Синтаксические и семантические ошибки. Методы и средства отладки программ. Тестирование программы методами "белого и черного ящиков". Подходы к построению тестовых задач. Планирование ресурсных затрат на тестирование.

Понятие стиля программирования. Структурное программирование. Модульная структура программы. Комментарии в программе. Подходы к выбору идентификаторов. Понятие дружелюбности интерфейса.

Оценка потребительских качеств программного продукта. Затраты вычислительных ресурсов. Тенденции развития информатики и программирования. Дисциплины информационного цикла в учебном плане Университета.

6. План подгрупповых (лабораторных) занятий

№ п/п	Наименование темы дисциплины	Наименование и содержание подгрупповых (лабораторных) занятий, литература для подготовки к занятиям	Формируемые компетенции	Формы контроля усвоения знаний
1.	Информация и информационные процессы	Информационный анализ процессов и явлений предметной области. Литература: 1-5.	УК-1, ОПК-1	Устный опрос Проверка файлов общих и индивидуальных заданий, устный опрос, оперативное исправление ошибок
2.	Программные средства обработки информации	Лабораторные работы на темы: 1. Создание и оформление сложных документов. 2. Профессиональная работа в электронных таблицах. Литература: 1-5.	ОПК-1, ОПК-4	Устный опрос Проверка файлов общих и индивидуальных заданий, устный опрос, оперативное исправление ошибок
3.	Постановка и алгоритмизация прикладных задач	Цикл лабораторных работ по алгоритмизации и программированию на тему: Проектирование,	ОПК-4, ОПК-7	Проверка индивидуальных заданий
4.	Методы и языки программирования	программирование, отладка и документирование программ - комплекс работ по программированию алгоритмических конструкций на языке высокого уровня. Литература: 1-5.	ОПК-4, ОПК-7	Проверка индивидуальных заданий

7. Образовательные технологии

При проведении учебных занятий по дисциплине для успешного освоения применяются различные образовательные технологии, которые обеспечивают развитие навыков командной работы, межличностной коммуникации, принятия решений, лидерских качеств.

Методы / Формы	Лекции (Л)	Семинарские занятия (С)
Диалого-дискуссионное обсуждение проблем	+	+
Работа в команде		+
Поисковый метод	+	+
Проектный метод		+

8. План самостоятельной работы студентов

№ п/п	Содержание самостоятельной работы студентов	Формируемые компетенции	Форма отчетности студента
1	Изучение литературы по теме	УК-1, ОПК-1, ОПК-4, ОПК-7	Представления обоснованной и развернутой структуры работы
2	Подготовка к практическому занятию	УК-1, ОПК-1, ОПК-4, ОПК-7	Представление выполненных работ
3	Выполнение заданий лабораторных работ	УК-1, ОПК-1, ОПК-4, ОПК-7	Отчеты о выполнении лабораторных работ
4	Выполнение дополнительных заданий для самостоятельной работы	УК-1, ОПК-1, ОПК-4, ОПК-7	Отчеты о выполнении самостоятельных работ, тест-опрос

9. Контроль знаний по дисциплине

По дисциплине предусмотрены текущий контроль и промежуточная аттестация.

Текущий контроль успеваемости студента – одна из составляющих оценки качества усвоения образовательных программ. Текущий контроль проводится в течение семестра в результате проверки выполнения циклов лабораторных работ.

Промежуточная аттестация проводится по окончании изучения дисциплины в виде зачета и экзамена. Вопросы к промежуточной аттестации сформулированы в **Оценочных и методических материалах**.

10. Учебно-методическое и информационное обеспечение дисциплины:

а) Основная литература

1. Зыков С. В. Программирование : учебник и практикум для академического бакалавриата / С. В. Зыков. — М. : Издательство Юрайт, 2019. — Режим доступа: <http://urait.ru/bcode/433432>
2. Кувшинов Д. Р. Основы программирования : учебное пособие для вузов / Д. Р. Кувшинов. — М. : Издательство Юрайт, 2019 ; Екатеринбург : Изд-во Урал. ун-та. — Режим доступа: <http://urait.ru/bcode/441475>
3. Трофимов В. В. Алгоритмизация и программирование : учебник для академического бакалавриата / В. В. Трофимов, Т. А. Павловская ; под ред. В. В. Трофимова. — М. : Издательство Юрайт, 2019. — Режим доступа: <http://urait.ru/bcode/423824>
4. Хлебников А. А. Информационные технологии : учебник / Хлебников А.А. — М. : КноРус, 2018. — Режим доступа: <http://book.ru/book/927689>
5. Черпаков И. В. Основы программирования : учебник и практикум для прикладного бакалавриата / И. В. Черпаков. — М. : Издательство Юрайт, 2019. — Режим доступа: <http://urait.ru/bcode/433423>

б) Дополнительная литература:

1. Гниденко И. Г. Технологии и методы программирования : учебное пособие для прикладного бакалавриата / И. Г. Гниденко, Ф. Ф. Павлов, Д. Ю. Федоров. — М. : Издательство Юрайт, 2019. — Режим доступа: <http://urait.ru/bcode/433611>
2. Казанский А. А. Программирование на Visual С# : учебное пособие для вузов / А. А. Казанский. — М.: Издательство Юрайт, 2019. — Режим доступа: <http://urait.ru/bcode/447372>
3. Нагаева И. А. Программирование: Delphi : учебное пособие для академического бакалавриата / И. А. Нагаева, И. А. Кузнецов ; под редакцией И. А. Нагаевой. — М. : Издательство Юрайт, 2019. — Режим доступа: <http://urait.ru/bcode/444273>
4. Огнева М. В. Программирование на языке С++: практический курс : учебное пособие для бакалавриата и специалитета / М. В. Огнева, Е. В. Кудрина. — М. : Издательство Юрайт, 2019. — Режим доступа: <http://urait.ru/bcode/438987>
5. Подбельский В. В. Программирование. Базовый курс С# : учебник для бакалавриата и специалитета / В. В. Подбельский. — М. : Издательство Юрайт, 2019. — Режим доступа: <http://urait.ru/bcode/439068>
6. Тузовский А. Ф. Объектно-ориентированное программирование : учебное пособие для прикладного бакалавриата / А. Ф. Тузовский. — М. : Издательство Юрайт, 2019. — Режим доступа: <http://urait.ru/bcode/434045>

г) Лицензионное программное обеспечение

1. Семейство программ Microsoft Office Standart Russian (Включает набор продуктов: Word, Excel, PowerPoint, Publisher, Outlook);
2. Mirapolis Virtual Room;
3. Антиплагиат;
4. КонсультантПлюс
5. Project Expert 7
6. Prime Expert
7. FineModel Expert
8. Обеспечено доступом к сети «Интернет» и электронной информационно-образовательной среде СПбГУП.

д) Современные профессиональные базы данных и информационные справочные системы

1. Официальный сайт СПбГУП: <http://www.gup.ru/>
2. Электронно-библиотечная система СПбГУП,
3. Системы поддержки самостоятельной работы СПбГУП: <http://edu.gup.ru/>
4. Справочная правовая система «Консультант плюс» <http://www.consultant.ru>
5. Российское образование <http://www.edu.ru/>
6. Единое окно доступа к образовательным ресурсам <http://window.edu.ru/>
7. Электронно-библиотечная система <http://e.lanbook.com/>

11. Материально-техническое обеспечение дисциплины

Аудиторный фонд с демонстрационным оборудованием и техническими средствами обучения, учебно-наглядные пособия и методические ресурсы кафедры, фонды Научной библиотеки.

Изучение дисциплины инвалидами и обучающимися с ограниченными возможностями здоровья осуществляется с учетом особенностей психофизического развития, индивидуальных возможностей и состояния здоровья обучающихся.

УЧЕБНО-МЕТОДИЧЕСКОЕ ОБЕСПЕЧЕНИЕ САМОСТОЯТЕЛЬНОЙ РАБОТЫ СТУДЕНТОВ

1. Методические рекомендации по организации самостоятельной работы студентов

Самостоятельная работа в высшем учебном заведении является важной организационной формой индивидуального изучения студентами программного материала. Эти слова особенно актуальны в наше время, когда в педагогике высококвалифицированных специалистов широко используется дистанционное обучение, предполагающее значительную самостоятельную работу студента на основе рекомендаций преподавателя.

2. Методические рекомендации по подготовке к практическим (подгрупповым) занятиям

Подгрупповые занятия — важная форма учебного процесса. Они способствуют закреплению и углублению знаний, полученных студентами на лекциях и в результате самостоятельной работы над научной и учебной литературой и нормативными источниками. Они призваны развивать самостоятельность мышления, умение делать выводы, связывать теоретические положения с практикой. На занятиях вырабатываются необходимые каждому специалисту навыки и умения, логика, культура профессиональной речи. Кроме того, семинары — это средство контроля преподавателей за самостоятельной работой студентов, они непосредственно влияют на уровень подготовки к итоговым формам отчетности — зачетам и экзаменам. В выступлении на занятии должны содержаться следующие элементы:

- четкое формулирование соответствующего теоретического положения в виде развернутого определения;
- приведение и раскрытие основных черт, признаков, значения и роли изучаемого явления или доказательства определенного теоретического положения;
- подкрепление теоретических положений конкретными фактами.

Для качественного и эффективного изучения дисциплины необходимо овладение навыками работы с книгой, воспитание в себе стремления и привычки получать новые знания из научной и иной специальной литературы. Без этих качеств не может быть настоящего специалиста ни в одной области деятельности.

Читать и изучать, следует, прежде всего, то, что рекомендуется к каждой теме программой, планом семинарских занятий, перечнем рекомендуемой литературы.

Когда студент приступает к самостоятельной работе, то он должен проявить инициативу в поиске специальных источников. Многие новейшие научные положения появляются, прежде всего, в статьях, опубликованных в журналах.

Надо иметь в виду, что в каждом последнем номере издаваемых журналов публикуется библиография всех статей, напечатанных за год, это облегчает поиск нужных научных публикаций.

Для качественного и эффективного изучения дисциплины необходимо овладение навыками работы с книгой, воспитание в себе стремления и привычки получать новые знания из научной и иной специальной литературы. Без этих качеств не может быть настоящего специалиста ни в одной области деятельности.

Читать и изучать, следует, прежде всего, то, что рекомендуется к каждой теме программой, планом семинарских занятий, перечнем рекомендуемой литературы.

Когда студент приступает к самостоятельной работе, то он должен проявить инициативу в поиске специальных источников. Многие новейшие научные положения появляются, прежде всего, в статьях, опубликованных в журналах.

Надо иметь в виду, что в каждом последнем номере издаваемых журналов публикуется библиография всех статей, напечатанных за год, это облегчает поиск нужных научных публикаций.

Работа с научной литературой, в конечном счете, должна привести к выработке у студента умения самостоятельно размышлять о предмете и объекте изучения, которое должно проявляться:

- в ясном и отчетливом понимании основных понятий и суждений, содержащихся в публикации, разработке доказательств, подтверждающих истинность тех или иных положений;
- в понимании студентами обоснованности и целесообразности, приводимых в книге и статье примеров, поясняющих доказательства и выводы автора. При этом будет уместно, если студент самостоятельно приведет дополнительные примеры к этим выводам;
- в отделении основных положений от дополнительных, второстепенных сведений;
- в способности студента критически разобраться в содержании публикации, определить свое отношение, дать общую оценку, характеристику.

Самостоятельная работа студентов предполагает тщательное освоение студентами учебной и научной литературы по изучаемой дисциплине. Изучение научной литературы – это серьезная работа, которую следует проводить по этапам:

- общее ознакомление с источником в целом по его оглавлению;
- беглый просмотр всего содержания;
- последовательное чтение материала;
- выборочное чтение какой-либо части текста;
- выписка материала, представляющего интерес;
- критическая оценка выписанного материала, его сравнение с другими точками зрения.

При самостоятельном изучении *основной рекомендуемой литературы* студентам необходимо обратить внимание на выделение основных понятий, их определение, узловых положений, представленных в изучаемом тексте.

В качестве информационно-справочного материала можно использовать глоссарий, в котором содержится перечень и определение ключевых понятий документооборота.

Нужно обратить внимание на образно-схематическое представление излагаемого материала в виде рисунков, схем, графиков и диаграмм, присутствующих в изучаемом тексте. Они способствуют более быстрому восприятию и запоминанию учебного материала.

Для контроля усвоения содержания темы в конце соответствующих глав и параграфов учебников и учебных пособий обычно дается перечень контрольных вопросов, на которые студент должен уметь дать четкие и конкретные ответы.

Самостоятельная работа с дополнительной литературой предполагает умение студента выделять в ней необходимый аспект изучаемой темы (сто в данном материале относится непосредственно к изучаемой теме и основным вопросам). к дополнительной литературе как правило относят широкий спектр текстов (учебных, научных, научно-популярных, художественных, публицистических и др.), в которых изучаемых вопрос рассматривается частично либо с нетрадиционной точки зрения.

Знакомство с дополнительной литературой, несомненно, обогащает знания и расширяет научный кругозор студентов.

Дополнительную литературу нужно обрабатывать после основной литературы и исследовать ее надо комплексно, рассматривая различные точки зрения на исследуемый вопрос.

При самостоятельном освоении информационных технологий и работе с применением программных продуктов следует использовать методическую литературу, содержащую детальное описание последовательности и технологии выполнения базовых операций.

Для закрепления теоретических знаний и практических навыков рекомендуется отрабатывать технологии на контрольном примере.

Общими правилами самостоятельного изучения элементов курса являются:

- изучение теоретических основ соответствующих разделов программы, изложенных в лекционном материале и литературных источниках;
- обучение на контрольных примерах (с заранее известным результатом – правильным ответом);
- эффективное использование системы помощи (HELP) применяемого программного обеспечения;
- заданная последовательность в изучении и освоении курса («от простого к сложному» — от отдельных элементарных операций к комплексным);
- использование индивидуальных заданий, уменьшающих вероятность списывания недобросовестными студентами;
- подготовка к ответам на контрольные вопросы.

Основные виды самостоятельной работы студентов:

1. **Предварительная подготовка к занятиям, в том числе и к тем, на которых будет изучаться новый материал.** Такая подготовка предполагает изучение учебной программы, установление связи с ранее полученными знаниями, выделение наиболее значимых и актуальных проблем, на изучение которых следует обратить особое внимание, ознакомление с материалом учебника. Также рекомендуется работа со словарями по новым экономическим терминам.

2. **Прислушивание и восприятие лекций**, что предполагает осмысление учебной информации, сообщаемой преподавателем, ее обобщение и краткую запись, своевременную доработку конспектов лекций. Основная задача студентов на лекционных занятиях – выделить главное в изучаемом материале, а также уяснить связь данной темы с логикой дисциплины в целом и предыдущих лекций. На лекционных занятиях приветствуются уточняющие вопросы со стороны студентов.

3. **Подбор, изучение, анализ и конспектирование рекомендуемой литературы.** Данная форма самостоятельной работы позволяет углубить знания в рамках определенной темы учебной программы. Основные рекомендации для эффективного освоения дополнительного материала: 1) до чтения: осмысление заголовка, анализ оглавления и аннотации; 2) по ходу чтения: старайтесь понять все слова и предложения в тексте, выделить важные и ключевые из них; задавайте вопросы самому себе по содержанию прочитанного, стройте свои предположительные ответы и сверяйте их с текстом; по ходу чтения попытайтесь осознать, что Вам не понятно, в чем возникают сомнения; делайте выписки, выделяйте главные мысли, составляйте схемы, графики, таблицы; 3) после прочтения: сформулируйте главную мысль прочитанного; прочитайте повторно наиболее сложные для Вас части текста; задайте вопросы себе по всему тексту; составьте конспект прочитанного; через время проверьте себя. Предварительное изучение рекомендуемой литературы позволяет отобрать необходимую учебную информацию и выяснить, по каким вопросам следует подобрать дополнительные литературные источники.

4. **Подготовка к семинарским, практическим занятиям и зачету.** Подготовку к семинару и практическому занятию лучше начинать сразу же после лекции по данной теме. Прежде всего, следует доработать текст лекции по соответствующей теме, внимательно изучить план семинара (практического занятия), содержания основных учебных вопросов, выносимых для обсуждения, а также список рекомендованной литературы и дополнительные задания, которые могут быть даны преподавателем. Далее необходимо подобрать и просмотреть литературу, которая рекомендована для подготовки к занятию. Это даст возможность выбрать те источники, где имеются ответы на поставленные учебные вопросы. Затем более внимательно ознакомиться с содержанием книги или статьи, отметить те части текста, в которых вопросы семинара раскрываются наиболее глубоко и подробно. В ходе углубленного чтения выделяются главные мысли, составляются краткие конспекты, тезисы своих будущих выступлений. Конспекты, которые отражают содержание вопросов семинарского и практического занятия, лучше

всего вести в той же тетради, в которой конспектируются лекции по дисциплине. Концентрация всей информации по проблеме в одном месте позволяет студентам активно участвовать в обсуждении вопросов семинара. В дальнейшем такие записи значительно облегчат подготовку к зачету. Подготовка к семинарским, практическим занятиям, зачету не сводится только к поиску ответов на поставленные вопросы. Любая теоретическая проблема должна быть осмыслена с точки зрения ее связи с реальной жизнью и возможностью реализации на практике. По каждому вопросу студент должен быть готов высказать и свою точку зрения.

Подготовка к семинарским, практическим занятиям, зачету не сводится только к поиску ответов на поставленные вопросы. Любая теоретическая проблема должна быть осмыслена с точки зрения ее связи с реальной жизнью и возможностью реализации на практике. По каждому вопросу студент должен быть готов высказать и свою точку зрения. Рекомендации по темам следующие.

Информация и информационные процессы

Понятие информации. Системы счисления и кодирование информации. Количественные оценки объема информации. Основные требования к компьютерным данным. Информационный процесс и его основные стадии. Понятие об информационном моделировании. Понятие системы, контур управления и информационные потоки в системе. Основные направления защиты информации. Базовые понятия и операции булевой алгебры и их использование в информационных системах.

При изучении этой темы обратите особое внимание на место теории информации в системе наук и влияние информатики на другие научные дисциплины. Важно практиковаться в построении информационных моделей для систем различной природы. Одним из ключевых для изучения этой темы является понятие количества информации (по К. Шеннону).

Контрольные вопросы

1. Предмет научной информатики. Информация как базовое понятие науки. Подходы к определению понятия «информация».

2. Понятие об информационном обществе. Основные признаки.

3. Признаки классификации информационных моделей. Построение реляционной, сетевой и иерархической моделей ситуации.

4. Количественные оценки объема информации. Понятие о вероятностном подходе к оценке объема информации с точки зрения ее содержания.

5. Формализация классификационных процессов. Понятие об иерархическом, фасетном и дескрипторном методах классификации.

6. Цели и методы сжатия информации.

7. Информационный процесс, его основные стадии и участники.

8. Понятие системы. Закрытые и открытые системы. Контур управления и информационные потоки в системе.

Программные средства обработки информации

Классификация операционных систем. Основные группы прикладных программ. Файловые системы. Альтернативность процедур выполнения операций в операционной среде.

Унифицированные функции интерфейса прикладных программ. Способы организации обмена данными между прикладными программами. Связь и внедрение объектов. Настройка операционной системы. Краткая характеристика функций сервисных программ. Файловые менеджеры. Командные файлы и макросредства. Способы защиты информации при ее хранении и обработке на ПК. Восстановление информации. Защита информации от несанкционированного доступа. Архивация программ и данных. Способы защиты данных от компьютерных вирусов.

Работа с текстовым редактором. Редактирование текстовых документов: шрифтовое и стилевое оформление текста. Дополнительные возможности (таблицы, формы, составные документы).

Работа с электронными таблицами. Применение математических, статистических и экономических функций. Работа с электронной таблицей как с базой данных. Дополнительные возможности.

Представление графической информации, форматы файлов. Типовые приемы обработки изображений.

При изучении данной темы большое место занимают самостоятельные практические занятия, поэтому для работы следует выбрать те возможности приложения, технология применения которых наиболее ясна и понятна для вас.

Контрольные вопросы

1. Понятие программного обеспечения. Классификация программных средств.
2. Структура системного программного обеспечения.
3. Совместимость прикладных программ с операционной средой.
4. Программные средства управления файлами и каталогами.
5. Понятие программного продукта. Характеристики пакетов прикладных программ.
6. Требования к пользовательскому интерфейсу.
7. Авторское право на программные продукты.
8. Защита программных продуктов.

Постановка и алгоритмизация прикладных задач

Содержательная постановка прикладной задачи. Критерии отбора значимых факторов и связей. Модель задачи. Формализованная постановка задачи, ее логико-математическая модель. Этапы решения: разработка алгоритма; проектирование программы; запись на языке высокого уровня; отладка; тестирование; серийное использование. Итеративность процесса создания программы. Пользовательский интерфейс программы. Понятие программного продукта. Авторское право на компьютерные программы.

Алгоритм и его свойства. Алгоритмические системы и машины Тьюринга. Частные и общие критерии сравнения алгоритмов. Представление алгоритмов в текстовой форме, в форме псевдокода и в виде блок-схемы. Использование готовых блоков. Основные алгоритмические конструкции: линейная, циклическая, разветвляющаяся. Рекуррентные алгоритмы. Понятие вычислительной сложности задач. Фиксация авторских прав на компьютерные алгоритмы.

Данная тема является определяющей при формировании алгоритмического мышления и при планировании самостоятельной работы должна рассматриваться совместно с темой 5.

Контрольные вопросы

1. Основные стадии компьютерного решения прикладной задачи.
2. Общее понятие алгоритма. Свойства компьютерных алгоритмов.
3. Словесная форма представления алгоритмов. Блок-схемы.
4. Основные алгоритмические конструкции: следование, цикл, переход, ветвление.
5. Понятие вычислительной сложности задач.
6. Классы алгоритмов с точки зрения вычислительной сложности.
7. Рекуррентные и итерационные алгоритмы.

Методы и языки программирования

Эволюция технологий программирования. Стадии преобразования программных модулей. Понятие «системы программирования». Процесс трансляции программ. Компиляторы и интерпретаторы. Классификация языков программирования. Основные

компоненты языка программирования: алфавит; словарь; набор операций; выражения; операторы; система синтаксических правил. Семантика программы.

Основные элементы программы: описания данных; описания процедур; цепочки операторов; тексты процедур; внутренние комментарии. Последовательность размещения элементов программы. Использование библиотек готовых программных модулей. Классификация обрабатываемых данных по типу. Способы описания типов. Понятие стандартных типов языка программирования и типов пользователя.

Программирование основных алгоритмических конструкций. Программирование ввода/вывода. Ввод/вывод структурированных данных. Программирование работы с графическими объектами. Программирование диалога с пользователем.

Классификация ошибок в программе. Подходы к формальному доказательству правильности программ. Синтаксические и семантические ошибки. Методы и средства отладки программ. Тестирование программы методами «белого и черного ящиков». Подходы к построению тестовых задач. Планирование ресурсных затрат на тестирование.

Понятие стиля программирования. Структурное программирование. Модульная структура программы. Комментарии в программе. Подходы к выбору идентификаторов. Понятие дружелюбности интерфейса.

В отличие от материала темы 3 для тем 4 и 5 лабораторных работ никогда не бывает достаточно. Поэтому следует использовать любую возможность для самостоятельной практики в написании программ. Большую пользу приносит и изучение текстов готовых программ, которое, впрочем, не может заменить самостоятельного программирования.

Контрольные вопросы

1. Понятие программирования для ЭВМ. Эволюция технологий программирования.
2. Классификация языков программирования. Подходы к выбору языка. Версии языка.
3. Основные компоненты языка программирования: алфавит; словарь; набор операций; выражения; операторы; синтаксис и семантика.
4. Назначение процедур. Правила описания процедур и обращения к ним.
5. Назначение, правила оформления и требования к содержанию комментариев.
6. Понятие пользовательского интерфейса диалоговых программ.
7. Присваивание значений переменным. Структура оператора присваивания. Понятие совместимости типов.
8. Подходы к оптимизации программ. Ситуации с повторяющимися подвыражениями. Понятие о чистке циклов.
9. Типичные семантические ошибки при программировании.
10. Основные подходы к тестированию программ.
11. Авторское право на компьютерные программы. Оформление законченной программы как программного продукта.

Содержание практических (семинарских) занятий

1. Рекомендации по выполнению индивидуальных заданий лабораторных работ

Лабораторная работа

«Применение информационного подхода к анализу процессов и явлений»

Для создания отчета рекомендуется использовать программный продукт MS Word, при составлении таблиц, для нормализации значений факторов и коэффициентов значимости следует применять MS Excel. Возможно также применение более мощных программных средств для автоматизации математических расчетов.

Лабораторная работа

«Создание и оформление сложных документов»

Для успешного выполнения заданий рекомендуется самостоятельно проработать следующий материал.

Варианты создания документа. Редактирование документов. Вставка специальных элементов в текстовый документ (сноски, примечания, аннотации, специальные символы и др.).

Форматирование текстовых документов. Стилизовое оформление. Применение организатора стилей. Создание и использование шаблонов документов. Вставка в текстовый документ оглавления, указателей, списка иллюстраций.

Создание и редактирование таблиц. Вычисления в текстовом редакторе, использование вычисляемых полей и формул.

Операции с графическими объектами и многослойными рисунками. Подготовка больших документов к печати: выделение разделов, нумерация страниц, формирование колонтитулов, предварительный просмотр. Навигация в текстовых документах с помощью закладок, переходов, гиперссылок.

Лабораторная работа

«Профессиональная работа в электронных таблицах»

Для успешного выполнения заданий рекомендуется самостоятельно проработать следующий материал.

Создание электронных таблиц. Вставка, переименование и удаление листов таблицы. Создание формул с абсолютными и относительными адресами, ссылками на другие листы, с функциями (ЕСЛИ, СУММ, МИН, МАКС и т. д.), с функциями для работы с таблицей как с базой данных (БДСУММ, БСЧЕТ, ДМАКС и т. д.) Создание имен для ячеек и применение их в формулах.

Форматирование и оформление таблицы. Применение форматов для значений ячеек типа «число» и «дата».

Построение и модификация диаграмм разных типов. Элементы оформления диаграмм и их изменение.

Организация фрагмента таблицы как базы данных, сортировка и поиск информации. Области критериев, формулы с критериями.

Решение с помощью электронных таблиц задач «Как сделать, чтобы?» и анализа чувствительности. Применение операций «Подбор параметра», «Таблица подстановки» и «Поиск решения», использование сценариев.

Цикл работ

«Основы алгоритмизации и программирования»

Все задания построены методически однотипно: вначале предлагается разобрать готовое решение задачи, набрать и отладить одинаковую для всех студентов группы задачу; затем — самостоятельно решить несколько однотипных задач по индивидуальному заданию. Данный подход позволяет устранить зависимость сроков выполнения заданий от индивидуальных способностей студента к программированию. Как общие, так индивидуальные задания студент может выполнять самостоятельно. В последнем случае на преподавателя возлагается дополнительная ответственность по контролю самостоятельного выполнения заданий и проверке степени усвоения учащимся материалов соответствующей лабораторной работы.

3. Методические рекомендации по написанию контрольных работ

Контрольная работа (реферат) – это письменная работа на определенную тему, подготовленная на основе изучения учебной и научной литературы, статей, а также обобщения личных знаний и практического опыта. Контрольная работа по курсу «Информатика и программирование» представляет собой одну из форм самостоятельной работы студентов по изучению и усвоению наиболее сложных теоретических проблем.

При написании контрольной работы студенты должны показать умение работать с основной и дополнительной литературой, способность обобщать материал.

Основные требования к реферату (контрольной работе):

- высокий теоретический уровень;
- четкость определения понятий и категорий;
- выявление правовых аспектов рассматриваемой проблемы;
- аргументированность основных положений и выводов работы;
- обязательное использование фактического (в том числе статистического) материала;

Выбор темы – одна из наиболее ответственных составных частей разработки контрольной работы. Любая тема по курсу «Документирование управленческой деятельности», особенно в условиях современного реформирования экономики России, является актуальной. Следовательно, знания, которые студент приобретет при написании контрольной работы, принесут положительные результаты, как с научной, так и с практической точек зрения. Не исключено, что в процессе работы название контрольной работы в рамках избранной темы потребует уточнения. Но, в конечном счете, нужно соблюдать главное требование: название темы должно соответствовать общему направлению излагаемого материала, отражать его основные идеи.

Студент может предложить свою тему контрольной работы, если она соответствует тематике курса. В этом случае тему необходимо согласовать с преподавателем, ведущим данный курс.

План – это основа, костяк письменной работы. От правильного его составления во многом зависят структура, содержание, логическая связь ее частей. Одновременно план – важный показатель того, насколько глубоко студент изучил экономические произведения (первоисточники) и другой необходимый материал (дополнительную литературу), сумел ли он отобрать из всего этого главное, существенное. Разработка плана состоит из двух этапов: составления предварительного плана и составления окончательного плана. Составление окончательного плана осуществимо лишь на основе более или менее полного овладения материалом, уяснения, по крайней мере, главных вопросов темы.

В плане должны быть отражены: введение; 2-3 вопроса; заключение; список использованной литературы. Литература подбирается студентом самостоятельно. Кроме того, он может обратиться за рекомендациями к преподавателю, ведущему курс. Когда план составлен, студент приступает к глубокому изучению литературы с целью раскрытия содержания каждого из пунктов плана. В первую очередь изучаются первоисточники и основная литература по теме. Кроме того, необходимо использовать и дополнительную литературу (монографии, брошюры, журнальные статьи и т.п.).

Работа над текстом – это кропотливый, систематический труд студента. Написание хорошей контрольной работы требует от автора подлинно добросовестного отношения к учебе, принципиальности, честности. Как и любая творческая работа, написание реферата (контрольной работы) – это длительное размышление над избранной темой, терпеливая переделка первого варианта, неустанное его совершенствование.

Написание следует начинать с введения. Именно во введении студент дает общую характеристику самой темы, краткий обзор изученной им литературы, показывает степень научной разработки проблемы, цели, а также предполагаемый личный вклад в ее достижение. В этом случае автор, видя целое, более целеустремленно и квалифицированно работает над темой.

Поскольку объем реферата (контрольной работы) фиксируется определенным количеством страниц, то перед написанием текста их необходимо распределить по главам или параграфам, с учетом введения и заключения, определить сроки выполнения каждой из глав. Разумеется, такое планирование может быть только ориентировочным, но в нем заложено дисциплинирующее и мобилизующее начало.

Следует заранее подготовить себя к тому, что написанный текст при редактировании потребует немалых доработок и переделок, а зачастую и переписывания заново целых страниц и даже глав. Таким образом, первоначальный вариант текста может оказаться

лишь черновиком. Если хочешь научиться хорошо писать, научись хорошо думать! Неточное изложение является показателем того, что предмет мышления и самому автору недостаточно ясен. В свою очередь, правильное мышление зависит от знания предмета исследования. В конце каждой главы или параграфа целесообразно сделать небольшие выводы по наиболее важным аспектам главы. Выводы помогают плавно перейти к следующему вопросу. Формулировка выводов, а также предложений и рекомендаций по главам или параграфам не отменяет необходимости заключения по работе в целом. По существу, такое заключение – особая глава реферата (контрольной работы), в которой в обобщенном виде подводятся итоги всего исследования, формулируются общие выводы, авторские предложения, отмечаются вопросы, которые не нашли в исследовании окончательного решения и требуют дальнейшей разработки. Выводы должны быть отработаны так, чтобы словам было тесно, а мыслям просторно. Выводы декларативного характера, не вытекающие из содержания работы, снижают ее уровень.

Реферат (контрольная работа), представляя собой уже законченную текстуально работу, нуждается в обязательном авторском редактировании. Речь идет об упорядочении текста в соответствии с требованиями и правилами литературного языка и стиля, характером, назначением и направленностью проведенного исследования. Редактирование требует от автора: исключительного внимания и сосредоточенности; окончательного уяснения изучаемого вопроса; умения критически подойти к своему тексту; решительного устранения всего лишнего, мешающего четкому освещению вопроса (повторений, многословия, пространных формулировок); осуществления его лишь после полного написания текста; скромности (избегайте ссылок на самого себя и местоимений «я», «мною», «мой» и т.п.); уважения мнения других. В ходе творческих дискуссий необходимо придерживаться максимально аргументированных высказываний; научной ответственности и добросовестности.

Структурно контрольная работа состоит из следующих элементов:

1. Введение (актуальность темы и основные задачи работы).
2. Основная часть (состоит из двух – трех вопросов; в первом параграфе рассматриваются теоретические аспекты проблемы, выводятся определения понятий, дается общая характеристика состояния того или иного явления в современной экономической действительности; во втором параграфе анализируются перспективы, прогнозируется динамика рассматриваемых хозяйственных процессов, обосновываются узловые положения).
3. Заключение (формулируются основные выводы и определяется конкретное их значение для практики хозяйственных преобразований в современной российской экономике);
4. Список использованной литературы (перечень основной и дополнительной литературы).

Общий объем контрольной работы должен составлять 10-12 страниц компьютерного текста через 1,5 интервала. На втором листе контрольной работы (после титульного листа) приводится ее план. На последнем листе дается список использованной литературы в следующей последовательности: 1) документы: законы, указы, постановления правительства; 2) труды и выступления руководителей государства и правительства, деятелей законодательной и исполнительной властей; 3) монографии, брошюры, статьи; 4) статистические и справочные издания; 5) электронные источники.

Список использованной литературы должен быть оформлен в соответствии с требованиями государственного стандарта:

Представление контрольной работы

Выполненная работа сдается методистам заочного факультета университета в установленный учебным планом срок, где она регистрируется и передается на проверку преподавателю. После проверки контрольная работа возвращается студенту для ознакомления замечаниями и рецензией. В случае незачета работы студент должен внести

в нее все необходимые исправления и дополнения, вытекающие из замечаний преподавателя. Проверенная работа (со всеми исправлениями, дополнениями и первой рецензией) представляется студентом на повторную проверку.

Задания для написания контрольных работ (для заочной формы обучения).

Общее задание — освоить способы моделирования и анализа экономических задач средствами электронных таблиц, работу с электронной таблицей как с базой данных, стандартные функции для работы с базой данных, применение расширенного фильтра и итогов. Самостоятельно смоделировать и проанализировать задачу из индивидуального задания.

- База данных «Программное обеспечение предприятия».
- База данных «Техническое обеспечение предприятия».
- База данных «Оргтехника предприятия».
- База данных «Персонал».
- База данных «Интернет ресурсы по информатике».
- База данных «Интернет ресурсы по программированию».
- База данных «Интернет ресурсы по языку программирования».
- База данных «Интернет магазины программного обеспечения».
- База данных «Интернет магазины мультимедиа контента».

4. Методические рекомендации по написанию курсовой работы

Целью курсовой работы является получение студентами практических навыков в решении задач с применением компьютера. При выполнении курсовой работы студенты должны изучить и практически освоить правила и приемы:

- перехода от содержательной формулировки задачи к постановке ее в виде, учитывающем применение компьютера;
- разработки алгоритма решения задачи и представления его в виде графических блок схем;
- представления алгоритма в виде программы на определенном алгоритмическом языке;
- подготовки материалов, необходимых для отладки программы на компьютер;
- отладки и тестирования программы;
- оформления результатов решения и разработки технической документации к программе.

При выполнении курсовой работы студент должен разработать программу решения сформулированной в задании задачи, подготовить материалы для отладки и тестирования программы на компьютере, отладить программу и оформить отчетную документацию по выполненной работе.

Методическое обеспечение процесса выполнения курсовой работы представлено в электронном учебно-методическом комплексе университета и включает полнотекстовые учебники и справочники по языку программирования Паскаль, более 80 индивидуальных заданий на курсовую работу и пример оформления пояснительной записки

1. Требования к разрабатываемым программам

Разрабатываемая при выполнении курсовой работы программа решения задачи, сформулированной в задании, должна удовлетворять требованиям, приведенным ниже.

1. В программе должен быть предусмотрен ввод значений исходных данных из файла и вывод результатов в файл.
2. Форма ввода исходных данных должна быть удобной с точки зрения задания их пользователем, должна учитывать особенности задачи и обеспечивать максимально простое изменение отдельных значений.
3. В программе должен быть предусмотрен контрольный вывод исходных данных, следующий непосредственно за их вводом.
4. Должны быть предусмотрены контроль корректности введенных исходных данных и вывод сообщений при обнаружении некорректных значений.
5. Все предусмотренные в программе "аварийные" завершения ее работы (в случае возникновения исключительных ситуаций) должны сопровождаться выводом поясняющих сообщений.
6. В ходе работы программы должны выводиться промежуточные результаты, получаемые после выполнения каждого логически завершенного ее участка.
7. Вывод в файл группы значений однородных данных должен производиться после формирования выходного массива.
8. Форма вывода результатов в выходной файл должна быть удобной для восприятия и содержать текстовые пояснения в терминах решаемой задачи, заголовки и т.д.
9. Схемы алгоритмов и тексты программ должны быть разработаны с учетом принципов структурного программирования.

2. Оформление курсовой работы

Результаты выполнения курсовой работы оформляются в виде пояснительной записки, в которой должны быть отражены результаты работы, проделанной на каждом этапе разработки и отладки программы.

Пояснительная записка подготавливается в текстовом редакторе Word и распечатывается на стандартных листах бумаги формата А4 (210x297). Схемы, рисунки, чертежи выполняются с использованием соответствующих автофигур на листах бумаги того же формата. Допускается применение специализированных программных средств для создания схем, например MS Visio и др. Подобные фрагменты должны быть корректно импортированы в файл отчета – так, чтобы для их отображения не требовалась установка соответствующих программ.

Схемы алгоритмов оформляются в соответствии с ГОСТ 19.701-90. Весь материал пояснительной записки должен быть сброшюрован и снабжен титульным листом установленного в СПбГУП образца, с указанием номера задания, номера группы, фамилии студента и фамилии руководителя курсовой работы.

3. Вопросы, требующие согласования с руководителем курсовой работы

Выполнение курсовой работы должно проводиться в тесном контакте с руководителем курсового проектирования на практических занятиях и консультациях, где должны быть разрешены все возникающие вопросы и по отдельным этапам, и по всей работе в целом. Ряд вопросов, от решения которых в наибольшей степени зависит успешное выполнение работы, или в значительной степени влияющих на расход вычислительных ресурсов при разработке и запуске программ, требуют обязательного согласования с руководителем. К этим вопросам относятся:

- 1) формулировка конечной цели, которая должна быть достигнута при решении задачи, и пример решения, иллюстрирующий понимание задачи студентом;
- 2) перечень исключительных ситуаций, которые могут возникнуть при решении задачи на компьютер;
- 3) вид носителя и форма задания исходных данных;
- 4) варианты значений исходных данных, предназначенные для отладки и тестирования программы;
- 5) форма представления результатов работы программы.

4. Порядок выполнения курсовой работы

Работа над созданием программы начинается с момента получения задания и включает в себя следующие основные этапы:

- 1) содержательная постановка задачи (составление технического задания);
- 2) формализованная постановка задачи (разработка математической модели);
- 3) алгоритмизация;
- 4) разработка структуры данных;
- 5) программирование и отладка;
- 6) тестирование программы;

Ниже приведены назначение и содержание каждого этапа, а также рекомендации по их выполнению.

Этап 1. Содержательная постановка задачи

Исходная задача формулируется, как правило, специалистами в соответствующей предметной области. Поэтому перед разработкой программы программисту необходимо добиться полного понимания содержания задачи. При этом должны быть получены ответы на следующие вопросы:

- понятна ли используемая терминология и сущность задачи (что должна делать программа);
- какие объекты, являются исходными для достижения цели (что дано), все ли исходные данные имеются, все ли данные необходимы, нет ли противоречий;
- какие объекты являются результатами (что должно быть получено);
- какие сделаны допущения;
- какие ограничения на область применения программы;
- какие дополнительные требования к программе должны быть выполнены (временные характеристики, требуемый объем памяти, точность представления исходных данных и результатов, виды носителей данных, защита информации и т.д.).

Итоги выполнения данного этапа должны быть проиллюстрированы на примере, отражающем сущность и особенности решаемой задачи. Полезно сопроводить пример иллюстрациями (графиками, рисунками, схемами и т.п.).

Этап 2. Формализованная постановка задачи

Основным содержанием данного этапа является формализованное описание решаемой задачи и метода ее решения с помощью соответствующего математического аппарата.

Прежде, чем приступить к программированию, необходимо изучить или разработать методы решения поставленной задачи.

Как правило, для решения одной задачи существует несколько методов. Надо попытаться найти все или хотя бы некоторые из них, чтобы выбрать тот, который наиболее полно удовлетворяет требованиям к результатам и к программе. Для значительного числа задач может не существовать быстрого метода нахождения точного решения. Переборные методы допустимы только для задач ограниченной размерности. Возможность нахождения приближенных решений следует оговорить с руководителем.

При изучении уже существующего метода следует обратить внимание на следующие его особенности:

- расчетные формулы;
- характеристики метода по скорости, точности, требуемой памяти и области его применимости.

Исходя из того, что важнее для конкретного случая: скорость, точность или память, выбирается подходящий метод.

Выбрав (разработав) метод решения задачи, следует переформулировать ее в терминах соответствующего математического аппарата:

- определить и описать математическую форму представления исходных данных и конечных результатов;
- сформулировать метод решения задачи, т.е. выбрать необходимые преобразования, численные методы, формулы, определяющие правила получения результатов по исходным данным.

Этап 3. Алгоритмизация

Алгоритмизация - это разработка алгоритма решения задачи, т.е. формулировка точного предписания, по которому должен осуществляться вычислительный процесс.

Для получения алгоритма необходимо выполнить следующие действия:

- 1) Преобразование формул к виду, удобному для алгоритмизации.
- 2) Проектирование схемы алгоритма. Алгоритм перед программированием целесообразно записать в виде псевдокода или блок-схемы.

Схемы должны отражать процесс разработки алгоритма методом пошаговой детализации с учетом принципов структурного программирования.

Схема первого (самого высокого) уровня составляется из укрупненных блоков, каждый из которых изображает типовой процесс обработки данных на компьютер (например, блок ввода данных, блок проверки правильности ввода данных, блок вычисления значений функции и т.п.). Каждому следующему уровню детализации алгоритма должна соответствовать схема, сопровождаемая указаниями на связь ее блоков с детализируемыми блоками предыдущего уровня.

Степень детализации на самом низком уровне должна быть такой, чтобы действия, соответствующие каждому блоку, могли быть легко записаны с помощью одного или небольшого числа предложений языка программирования.

В процессе детализации схемы алгоритма должны быть выделены участки программы, которые целесообразно оформить в виде отдельных подпрограмм (процедур или функций).

При разработке схемы алгоритма следует учитывать следующие правила:

- схема выполняется с учетом требований ГОСТ 19.701-90. На каждую подпрограмму и головную программу составляется отдельная схема;

- схема должна быть читаема без дополнительных пояснений автора, поэтому схемы алгоритмов должны быть снабжены достаточным для их понимания количеством комментариев;

- степень подробности схемы должна быть такой, чтобы она полностью размещалась на стандартном листе бумаги формата А4;

- управление в схеме должно передаваться в основном вниз, возвраты допускаются только в циклах. Альтернативно выполняемые ветви должны размещаться параллельно;

- схема алгоритма должна состоять только из основных управляющих структур типа "следование", "ветвление" и "цикл", для которых характерно наличие только одного входа и только одного выхода. Как следствие, каждая схема должна иметь только один блок "начало" и один блок "конец".

Обработка исключительных ситуаций. При составлении алгоритма следует предусмотреть поведение программы в исключительных ситуациях, т.е. ситуациях, в которых алгоритм может не давать результата. Такие ситуации могут возникать по следующим причинам:

- особенности конкретной задачи (например, система линейных уравнений не всегда имеет решение);

- ошибки во входных данных;

- нарушение пользователем установленного интерфейса.

Общим требованием к программам является то, что при любых исходных данных программа должна доработать до конца и выдать либо результат, либо сообщение о невозможности получить результат. Отсюда следует, что там, где может возникнуть исключительная ситуация, алгоритмом должны быть предусмотрены соответствующие проверки и действия по устранению или обходу этой ситуации.

Этап 4. Разработка структуры данных

На этом этапе конкретизируется представление данных в будущей программе.

Определяются структуры, типы и имена для объектов, используемых в программе с учетом диапазонов значений, требуемой точности представления исходных данных, промежуточных данных и результатов, а также с учетом особенностей используемого алгоритма.

Выбор имен является важным моментом в процессе решения задачи. Каждое имя должно в максимально возможной степени отражать назначение объекта и легко запоминаться. Следует избегать бессмысленных сокращений, использования трудно различимых комбинаций символов.

Необходимо учитывать, что структуры данных и алгоритмы оказывают друг на друга существенное влияние. В зависимости от выбора структуры данных программа может значительно меняться по размерам и времени выполнения. Выбор конкретного алгоритма во многих случаях определяет и требования к структуре данных. Поэтому структура данных уточняется и дополняется по мере детализации программы.

Этап 5. Программирование и отладка

Основное содержание этапа - получение исходного текста программы.

Кодирование выполняется с обеспечением следующих требований.

Обязательно наличие комментариев - заголовков программы с указанием ее структурных и функциональных характеристик, правил обращения к программе, данных об авторе (фамилия, инициалы и номер группы студента), даты создания.

Делаются также комментарии к операторам объявления переменных (для каждой переменной в отдельности);

При оформлении текста программы используются отступы, сдвиги и пропуски при формировании текста. Операторы размещаются по строкам - не более одного оператора в строке.

Применяются правила структурного программирования, используются структурные программные конструкции ("следование", "ветвление", "цикл"). Выполняется автономная отладка модулей, затем комплексирование и отладка программы в целом.

При планировании отладки определяются точки программы, где будут использованы средства отладки, и принимаются решения о том, какие средства будут применены. Отладочные средства ставятся в узловых точках программы, на входах в процедуры; на длинных линейных участках ставят операторы для печати промежуточных результатов. Если есть возможность, используются различные отладочные режимы системы.

Результатом выполнения данного этапа является исполняемый модуль программы (файл с расширением .exe).

Этап 6. Тестирование программы

Основное содержание - проверка соответствия программы требованиям, сформулированным в техническом задании.

В начале производится планирование процесса тестирования, разработка контрольных примеров (тестов).

Количество тестов должно быть таким, чтобы при испытаниях был обеспечен всесторонний контроль работы программы:

- проверка работы всех блоков не только для номинальных, но также для исключительных и предельных значений;
- проверка всех вариантов ввода и вывода, включая сообщения об ошибках;
- выполнение всех операторов и всех ветвей передачи управления.

Для тестирования методами белого ящика тесты сопровождаются таблицами, подтверждающими полноту тестирования по соответствующему методу, например в форме следующей матрицы, учитывающей покрытие тестами вариантов ветвлений в программном модуле.

Если результаты выполнения программы расходятся с ожидаемыми, то принимаются меры к поиску причин расхождения и устранению ошибок. Все внесенные изменения фиксируются вместе с указанием причины модификации программы. В данном случае требуются краткий перечень найденных на данном этапе ошибок.

5. Содержание пояснительной записки к курсовой работе

Пояснительная записка к курсовой работе должна содержать подробное описание процесса разработки и оформлена в соответствии с требованиями ГОСТ. Количество разделов курсовой работы должны совпадать с количеством этапов разработки программы. Разделам присваиваются заголовки, совпадающие с наименованиями этапов. Содержание каждого раздела должно отражать действия программиста и результаты, полученные на соответствующем этапе. Далее необходимо привести краткую характеристику программы, представляющая собой описание свойств программы, знание

которых необходимо пользователю для ее эксплуатации. В это описание включаются: наименование программы; назначение программы; формат вызова программы; форматы ввода исходных данных и вывода результатов; простейшие метрики программы (количество строк исходного текста программы, размер исполняемого модуля, оценки времени работы программы).

Пояснительная записка должна заканчиваться заключением, содержащим выводы о проделанной работе и оценкой работоспособности полученной программы.

ОЦЕНОЧНЫЕ И МЕТОДИЧЕСКИЕ МАТЕРИАЛЫ

Оценочные и методические материалы включают в себя:

- перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы;
- показателей и критериев оценивания компетенций на различных этапах их формирования, описание шкал оценивания;
- 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;
- методические материалы, определяющие процедуры оценивания знаний, умений, навыков и (или) опыта деятельности, характеризующих этапы формирования компетенций.

1. Перечень компетенций с указанием этапов их формирования в процессе освоения образовательной программы

№ п\п	Контролируемые темы дисциплины	Код формируемой компетенции	Код и наименование индикатора достижения	Наименование оценочного средства
	Информация и информационные процессы	УК-1, ОПК-1	<p>УК -1.1. Знает основные положения информатики как научной дисциплины, изучающей процессы обработки информации, принципы организации и функционирование современных средств обработки информации, базовые принципы информационного анализа процессов и явлений предметной области.</p> <p>УК -1.2. Умеет соотносить разнородные явления и систематизировать их в рамках избранных видов профессиональной деятельности.</p> <p>УК -1.3. Имеет практический опыт работы с информационными источниками, опыт научного поиска, создания научных текстов.</p> <p>ОПК-1.1. Знает основные положения информатики как научной дисциплины, изучающей процессы обработки информации, принципы</p>	Опрос, участие в коллоквиуме, выполнение практических заданий, кейсы

			организации и функционирование современных средств обработки информации, базовые принципы информационного анализа процессов и явлений предметной области; методы проектирования и разработки алгоритмов и программ ОПК-1.3. Владеет навыками информационного анализа процессов и явлений, составления алгоритмов.	
	Программные средства обработки информации	ОПК-1, ОПК-4	ОПК-1.2. Умеет решать задачи обработки данных; составлять алгоритмы, проектировать, программировать программы на одном из языков высокого уровня с использованием современных технологий программирования ОПК-1.3. Владеет навыками информационного анализа процессов и явлений, составления алгоритмов. ОПК-4.1. Знает основные положения информатики как научной дисциплины, изучающей процессы обработки информации, принципы организации и функционирование современных средств обработки информации, базовые принципы информационного анализа процессов и явлений предметной области; методы проектирования и разработки алгоритмов и программ.	Подготовка сообщений, выполнение практически х заданий, тестирование
	Постановка и алгоритмизация прикладных задач	ОПК-4, ОПК-7	ОПК-4.2. Умеет решать задачи обработки данных с помощью современных инструментальных средств конечного пользователя; составлять алгоритмы, проектировать, программировать и тестировать программы на одном из языков высокого уровня с использованием современных технологий программирования ОПК-4.3. Владеет навыками документирования программ на одном из языков высокого уровня с использованием современных программных средств ОПК-7.1. Знает принципы организации проектирования и содержание этапов процесса разработки программных комплексов дисциплины, задачи и методы исследования и обеспечения качества и надежности программных	Опрос, тестирование, коллоквиум, кейсовые задания

			компонентов, основы разработки программных комплексов. ОПК-7.2. Умеет проводить анализ и проектирование современного программного обеспечения, определять его качественные показатели, организовывать процесс разработки и вести документацию в соответствии с современными стандартами.	
	Методы и языки программирования	ОПК-4, ОПК-7	ОПК-4.2. Умеет решать задачи обработки данных с помощью современных инструментальных средств конечного пользователя; составлять алгоритмы, проектировать, программировать и тестировать программы на одном из языков высокого уровня с использованием современных технологий программирования ОПК-4.3. Владеет навыками документирования программ на одном из языков высокого уровня с использованием современных программных средств ОПК-7.1. Знает принципы организации проектирования и содержание этапов процесса разработки программных комплексов дисциплины, задачи и методы исследования и обеспечения качества и надежности программных компонентов, основы разработки программных комплексов. ОПК-7.2. Умеет проводить анализ и проектирование современного программного обеспечения, определять его качественные показатели, организовывать процесс разработки и вести документацию в соответствии с современными стандартами. ОПК-7.3. Владеет навыками программирования, отладки и тестирования программного обеспечения.	Опрос, тестирование, коллоквиум, практические задания
Результат достижения планируемых результатов изучения дисциплины				Зачет, экзамен

2. Описание показателей и критериев оценивания компетенций, шкал оценивания

Критерии оценивания (текущий контроль)

1. Оценка **«отлично»** выставляется студенту, если студент имеет глубокие знания учебного материала по теме практического задания, в логической последовательности излагает материал; смог ответить на все уточняющие и дополнительные вопросы;
2. Оценка **«хорошо»** выставляется, если студент показал знание учебного материала, смог ответить почти полностью на все заданные дополнительные и уточняющие вопросы;
3. Оценка **«удовлетворительно»** выставляется, если студент в целом освоил материал; однако, ответил не на все уточняющие и дополнительные вопросы;
4. Оценка **«неудовлетворительно»** выставляется студенту, если он имеет существенные пробелы в знаниях основного учебного материала по теме практического задания, который полностью не раскрыл содержание вопросов, не смог ответить на уточняющие и дополнительные вопросы.

Критерии оценивания (зачет)

Знания, умения, навыки и компетенции студентов оцениваются следующими оценками: «зачтено», «не зачтено».

- **«зачтено»** - студент хорошо и прочно усвоил весь программный материал, исчерпывающе, последовательно, грамотно и логически стройно его излагает, увязывает с практикой, свободно справляется с решением ситуационных задач и тестовыми заданиями, правильно обосновывает принятие решений, умеет самостоятельно обобщать программный материал, не допуская ошибок, знает дополнительную литературу по изучаемой дисциплине.
- **«не зачтено»** - студент не знает значительной части основного программного материала, в ответах допускает существенные ошибки, не владеет умениями и навыками в выполнении тестовых заданий и решении задач, не способен ответить на дополнительные вопросы.

Критерии оценивания (экзамен)

Знания, умения, навыки и компетенции студентов оцениваются следующими оценками: «отлично», «хорошо», «удовлетворительно», «неудовлетворительно»

Оценка	Критерии оценивания
Отлично	Студент не только глубоко и прочно усвоил весь программный материал, но и проявил знания, выходящие за его пределы, почерпнутые из дополнительных источников (классическая литература, учебная литература, научно-популярная литература, научные статьи и монографии и т. п.); умеет самостоятельно обобщать программный материал, не допуская ошибок, проанализировать его с точки зрения различных школ и взглядов; увязывает знания с практикой, приводит примеры, демонстрирующие глубокое понимание материала или проблемы, свободно справляется с задачами и практическими заданиями; исчерпывающе, последовательно, грамотно и логически стройно выстраивает свой ответ.
Хорошо	Студент твердо знает программный материал, грамотно и последовательно его излагает, увязывает с практикой, не допускает существенных неточностей в ответе на вопросы, может правильно применять теоретические положения и владеет необходимыми умениями и навыками в выполнении практических заданий и решении задач, испытывает незначительные затруднения при самостоятельном обобщении программного материала.
Удовлетворительно	Студент усвоил только основной программный материал, но не знает его отдельных положений, в ответе допускает неточности, недостаточно правильные формулировки, нарушает последовательность в изложении программного материала, не в полной мере владеет необходимыми умениями и навыками в выполнении практических заданий и решении задач, испытывает затруднения при самостоятельном обобщении программного материала.
Неудовлетворительно	Студент не знает значительной части основного программного материала, в ответе допускает существенные ошибки, неправильные формулировки, не владеет необходимыми умениями и навыками в выполнении практических заданий и решении задач, испытывает значительные затруднения при самостоятельном обобщении программного материала.

3. Типовые контрольные задания и методические материалы, процедуры оценивания знаний, умений и навыков

ТЕКУЩИЙ КОНТРОЛЬ

Тестовые материалы

Важными в методическом плане на семинарских занятиях являются проводимые тестовые задания, которые содействуют укреплению теоретических, позволяют сделать выводы об эффективности занятий с учащимися.

Решение тестовых заданий является важным методическим приемом для закрепления и осмысления, полученных бакалаврами знаний по изучаемому предмету.

Студент тестируемой учебной группы получает 100 тестовых заданий. Для каждого из вопросов тестового задания предусмотрен только один правильный вариант ответа, который должен выбрать студент. Результаты тестирования оцениваются в зависимости от количества неверно выбранных ответов.

Итоги тестирования заносятся в ведомость, составляемую на всю учебную группу. Предоставленные сведения должны содержать данные о количестве опрошенных, о количестве отличных, хороших, удовлетворительных и неудовлетворительных оценок.

В заключение работы выводится средний балл итогового контроля знаний студентов.

ПАСПОРТ ТЕСТОВЫХ ЗАДАНИЙ

1. Общее количество тестовых заданий в базе – 212
2. Ограничение времени выполнения теста (в мин) – 30
3. Автоматическое перемешивание вопросов в тесте: (да)
4. Случайный порядок ответов в тестовом задании: (да)
5. Критерии оценки результатов тестирования : свыше 50% правильных ответов – зачет

Пример тестовых заданий для текущего контроля представлен ниже:

- 1. Сведения (сообщения, данные) независимо от формы их представления – это**
 - Документ
 - Публикация
 - База данных
 - Информация
- 2. Условием включения информации в информационные ресурсы является**
 - Протоколирование
 - Сохранение
 - Документирование
 - Накопление
- 3. Информацию, достаточную для решения тех или иных задач, называют**
 - Понятной
 - Полезной
 - Актуальной
 - Полной
 - Достоверной
- 4. Информация в теории информации – это**
 - Сведения, полностью снимающие или уменьшающие существующую до их получения неопределенность
 - Сведения, обладающие новизной
 - Отраженное разнообразие
 - То, что поступает в наш мозг из многих источников и во многих формах и, взаимодействуя там, образует нашу структуру знания
 - Неотъемлемый атрибут материи
- 5. Информацию, не зависящую от чье-либо мнения или суждения, называют**
 - Достоверной
 - Актуальной

- Объективной
 - Понятной
 - Полезной
- 6. Кодом называется**
- Правило, описывающее отображение набора знаков одного алфавита в набор знаков другого алфавита
 - Произвольная конечная последовательность знаков
 - Правило, описывающее отображение одного набора знаков в другой набор знаков или слов
 - Двоичное слово фиксированной длины
 - Последовательность слов над двоичным набором знаков
- 7. Русский алфавит может быть закодирован с помощью двоичных слов постоянной длины различными способами. Среди этих способов, отличающихся друг от друга, прежде всего, длиной кода, есть те, для которых длина кодового слова минимальна. Эта длина равна:**
- 1
 - 6
 - 5
 - 8
 - 2
- 8. В соответствии с кодовой таблицей ASCII символы английского алфавита кодируются двузначными числами, причем сочетание «I LOVE» кодируется так 733276798669, а сочетание «I LIVE»:**
- 733276738669
 - 733279768669
 - 733276867669
 - 733273768669
 - 763273737686
- 9. Укажите самое большое число**
- $(756)_{13}$
 - $(756)_{12}$
 - $(756)_{16}$
 - $(756)_8$
 - $(756)_{10}$
- 10. Пиксель – это**
- Минимальный элемент растрового изображения
 - Минимальный элемент векторного изображения
 - 32-битное изображение геометрическая точка
- 11. Растровое изображение представлено совокупностью**
- Разноцветных линий
 - Графических примитивов
 - Разноцветных точек
 - Правильных треугольников
- 12. Ухудшение качества изображения, связанное с увеличением размеров, характерно для**
- Растровой графики
 - Векторной графики
 - Автофигур, созданных в Word
- 13. Характерной особенностью векторной графики является**
- Ухудшение качества изображения с уменьшением его размера
 - Ухудшение качества изображения с увеличением его размера

- Уменьшение размера изображения с улучшением его качества
- Неизменность качества изображения с увеличением его размера

14. Непечатаемые знаки

- Не печатаются на принтере
- Не отображаются в документе после его сохранения
- Показывают измененные места в документе
- Показывают места в документе, которые не будут отпечатаны

15. Пробел ставится

- До знака препинания
- До знака "дефис"
- После знака "дефис"
- После знака препинания

16. С помощью каких операций в текстовом редакторе Word можно в формуле выполнить умножение

- Использовать стандартную функцию PRODUCT
- Использовать стандартную операцию *
- Использовать стандартную функцию УМНОЖ
- Использовать стандартную функцию ROUND
- Использовать стандартную функцию PRODUCT

17. Использовать стандартные ключевые слова можно использовать в WORD

- LEFT
- CENTER
- RIGHT
- ABOVE
- UNDER

18. Как начать список в текстовом редакторе Word

- Набрать 1. Иванов и нажать Enter
- Набрать Иванов и нажать кнопку Нумерация
- Выбрать стиль нумерованный список
- Выбрать стиль номер строки
- Набрать Иванов и выполнить Формат - Список - Нумерованный - Ok, набрать Иванов

19. Как закончить список в текстовом редакторе Word

- Два раза нажать Enter
- Enter, затем Back Space
- Enter, затем Delete
- Enter и отключить список с помощью кнопки Нумерация
- Enter, Формат - Список - Нет - Ok

20. Текстовая последовательность в электронной таблице Excel может быть использована при заполнении

- Названий месяца
- Названий дней недели
- Названий кварталов
- Любой текстовой последовательности, если для последовательности создан список пользователя

21. Числовая последовательность в электронной таблице Excel используется для заполнения интервала

- С любым шагом
- С целыми числами

- С дробными числами
 - С отрицательными числами
 - С вычисляемым по формуле шагом
- 22. Данные, содержащиеся в ячейке в электронной таблице Excel, можно редактировать**
- В меню
 - В строке формул
 - В ячейке
 - В специальном окне
- 23. Принципиальное отличие в электронной таблице Excel круговой диаграммы от всех остальных типов диаграмм**
- Может отображать только один ряд данных
 - Может отображать только одну категорию значений
 - Может показать относительную долю значений в общем результате
- 24. Что лучше показывает совмещенная столбиковая диаграмма в электронной таблице Excel**
- Относительную долю значений в общем результате
 - Изменение суммы значений каждой категории
 - Изменение суммы значений каждой категории и относительную долю значений
- 25. Что лучше показывает диаграмма площадей в электронной таблице Excel**
- Относительную долю значений в общем результате
 - Тенденции в изменении значений каждого ряда и относительной доли значений
 - Изменение суммы значений каждой категории
 - Изменение суммы значений каждой категории и относительную долю значений
- 26. Адрес ячейки в Excel – это**
- Любая последовательность символов
 - Имя столбца и номер строки
 - Адрес ячейки оперативной памяти
 - Номер строки и номер столбца
- 27. Ячейка электронной таблицы может содержать**
- Только формулу
 - Числа, формулы и текст
 - Только числа и текст
- 28. Число 124,56E7, записанное в ячейке таблицы, имеет следующий математический эквивалент**
- 124,56000000
 - 1245600000
 - 124560000000
 - 0,000012456
- 29. Выберите верное обозначение строки в электронной таблице**
- 18D
 - K13
 - 34
 - AB1
- 30. В адресе R[4]C[-2] в электронной таблице Excel**
- Абсолютен номер столбца, номер строки относителен
 - Абсолютен номер строки, номер столбца относителен
 - Все адреса абсолютны

- Адрес содержит ошибку
 - Все адреса относительны
- 31. В адресе R2C[-2] в электронной таблице Excel**
- Абсолютен номер столбца, номер строки относителен
 - Абсолютен номер строки, номер столбца относителен
 - Все адреса абсолютны
 - Адрес содержит ошибку
 - Все адреса относительны
- 32. Имя ячейки используется в электронной таблице Excel в**
- Формулах
 - При автоматическом построении отчетов
 - Для обозначения рядов данных в диаграммах
 - Для применения форматов чисел
- 33. Если в ячейке электронной таблицы отображается последовательность символов #####, то это означает, что**
- Формула записана с ошибкой
 - В формуле есть ссылка на пустую клетку
 - В формуле есть циклическая ссылка
 - Столбец недостаточно широк
- 34. В каком из указанных диапазонов содержится ровно 20 клеток электронной таблицы**
- E12:F12
 - C12:D11
 - C3:F8
 - A10:D15
- 35. В ячейках таблицы записаны следующие числа A1=1, A2=3, A3=5 в B1 записана формула =ЕСЛИ(A1>2;A2;A3), в B2 записана формула =ЕСЛИ(A2>2;0;1), в B3 записана формула =СУММ(B1:B2) . Тогда значение в B3 будет равно**
- 6
 - 3
 - 4
 - 5
- 36. Какую функцию можно выбрать в электронной таблице Excel для нахождения наибольшего значения в диапазоне?**
- СЧЁТ
 - ABS
 - МАКС
 - РОСТ
- 37. Какая категория функций в электронной таблице Excel содержит функцию для расчета среднего арифметического значения?**
- Финансовые
 - Математические
 - Статистические
 - Проверка свойств и значений
- 38. Можно ли изменить стиль в электронной таблице Excel, в параметрах которого установлен флажок защита?**
- Нет
 - Да, но только после того, как защита будет снята
 - Да, но при этом нельзя изменять числовые форматы
 - Да

- 39. Как можно скрыть отображенное примечание к ячейке в электронной таблице Excel**
- Щелкнуть по примечанию правой кнопкой мыши и выполнить команду Скрыть примечание
 - Щелкнуть по ячейке правой кнопкой мыши и выполнить команду Скрыть примечание
 - Выделить ячейку и выполнить команду Окно/Скрыть
- 40. Метод конструирования алгоритма из базовых алгоритмических конструкций называется**
- Конструкционным
 - Композиционным
 - Структурным
- 41. Формальное доказательство правильности алгоритма называется**
- Трассировкой
 - Тестированием
 - Верификацией
- 42. При решении задачи правильно**
- Упрощать максимально алгоритм за счет усложнения данных
 - Упрощать максимально данные за счет усложнения алгоритма
 - Упрощать алгоритм за счет более адекватной структуры данных
- 43. Алгоритм не обладает свойством**
- Формализованности
 - Массовости
 - Индивидуальности
- 44. Существуют задачи для которых не возможно построить алгоритм**
- Да
 - Нет
 - Не доказано
- 45. Экспоненциальный алгоритм может работать быстрее полиномиального**
- Да
 - Нет
 - Не доказано
- 46. Существуют алгоритмы время работы которых при росте размерности задачи слабо зависит от быстродействия компьютеров**
- Да
 - Нет
 - Не доказано
- 47. Существуют алгоритмы основанные на использовании датчика случайных чисел**
- Да
 - Нет
 - Не доказано
- 48. Существуют задачи для которых не возможно построить приближенный алгоритм с гарантированной оценкой погрешности решения**
- Да
 - Нет
 - Не доказано
 - Не доказано, но скорее всего это так
- 49. Фрагмент вида $y := 0; x := 12345; \text{for } i := 1 \text{ to } 3 \text{ do begin } y := y + x \bmod 10; x := \text{int}(x/10) \text{ end};$ вычисляет значение переменной y , равное**

- 10
- 13
- 12
- 15
- 0

50. Фрагмент вида $k := 1; s := x[1]; \text{for } i := 1 \text{ to } n \text{ do if } (s < x[i]) \text{ then begin } s := x[i]; k := i \text{ end; write}(k)$; выводит значение переменной k , равное:

- Максимальному элементу массива чисел $x[1], x[2], \dots, x[n]$
- Сумме элементов $x[1], x[2], \dots, x[n]$, меньших, чем $x[1]$
- Индексу максимального элемента ряда $x[1], x[2], \dots, x[n]$

51. Фрагмент вида $k := 1; s := x[1]; \text{for } i := 1 \text{ to } n \text{ do if } (s < x[i]) \text{ then begin } s := x[i]; k := i \text{ end; write}(s)$; выводит значение переменной s , равное:

- Максимальному элементу массива чисел $x[1], x[2], \dots, x[n]$
- Сумме элементов $x[1], x[2], \dots, x[n]$, меньших, чем $x[1]$
- Индексу максимального элемента ряда $x[1], x[2], \dots, x[n]$

52. Правда ли, что

- Вызов подпрограммы занимает меньше процессорного времени, чем вызов процедуры
- При вызове подпрограммы выделяется область памяти для локальных переменных
- Параллельные процессы имеют общую область памяти

53. Правда ли, что

- Комментарии в исходном тексте облегчают сопровождение программы
- Управление качеством продукта - часть системы управления организации
- Отслеживание хода разработки проекта - проблема управления персоналом

54. Тестирование - это процесс исполнения программы

- Чтобы показать, что программа работает правильно
- С целью выявления ошибок
- С целью устранения ошибок

55. При тестировании проверяется

- Поведение программы при нормальных входных данных
- Поведение программы при неправильных входных данных
- Осмысленность сообщений об ошибках

56. Синтаксические ошибки в программе устраняют в процессе

- Отладки
- Сопровождения
- Тестирования

ПРОМЕЖУТОЧНАЯ АТТЕСТАЦИ

1. Вопросы для подготовки к промежуточной аттестации по дисциплине (зачету)
2. Информация как ресурс. Способы хранения и обработки информации. Этические и этнические конфликты в информационном обществе. Информационно-культурное пространство интернета и виртуальной реальности.
3. Понятие структурированных данных. Определение и назначение базы данных. Соблюдение кодексов профессиональной этики при использовании информационных ресурсов сети интернет.
4. Концепция информационного общества. Современные проблемы информатизации общества. Информационные процессы. Информационные системы. Этапы развития, примеры, классификации и структура информационных систем.

5. Современные информационные технологии. Понятия, основные определения, сущность информационных технологий. Сетевая коммуникация и распространение визуальной информации.
6. Содержательный и технологический аспекты процесса интернетизации. Ключевые тенденции развития медиакommunikаций и социальные сети. Риски, связанные с использованием Интернет-пространства.
7. Взаимодействие гуманитарной и технологической цивилизаций. Основные причины формирования единого информационного пространства.
8. Назначение, основные возможности текстового редактора Word. Запуск и завершение работы. Основные элементы окна. Справочная система: команды, окна, Мастер подсказок (помощник), контекстно-зависимая справка. Режимы просмотра документов.
9. Управление окном приложения и окном документа. Структура меню и кнопочных панелей инструментов текстового процессора. Настройка панелей инструментов и лент. Колонтитулы и сноски: назначение, виды, способы создания.
10. Понятия текстового файла и текстового документа. Создание текстовых файлов и текстовых документов. Основные элементы текстового документа: символы, слова, строки, предложения, абзацы. Структура страницы. Структура документа.
11. Операции форматирования символов и абзацев. Общие правила форматирования символов и абзацев. Копирование формата.
12. Основные операции с текстом. Перемещение по набранному тексту. Способы выделения объектов текстового документа. Поиск и замена. Удаление, перемещение и копирование объектов текстового документа. Вставка полного текста из другого файла. Вставка специальных символов из таблицы символов.
13. Понятие шрифта. Параметры шрифтов. Шрифты TrueType. Установка стандартных гарнитуры и кегля по умолчанию в шаблоне.
14. Основные элементы страницы текстового документа Word. Создание и редактирование колонтитулов. Вставка сносок, примечаний. Блокировка разрыва страницы между абзацами. Подгонка страницы. Вставка номеров страниц, даты создания документа. Вставка разрывов страницы.
15. Работа с таблицами. Способы создания таблиц. Выделение структурных элементов. Основные операции с таблицами: объявление первой строки таблицы заголовком; перемещение по ячейкам таблицы; вставка и удаление строк, столбцов; изменение ширины (высоты) столбца (строки). Форматирование таблицы.
16. Преобразование текстового документа в таблицу. Обрамление и заливка выделенных ячеек таблицы. Сохранение таблицы как элемента автотекста. Управление расположением таблицы на странице документа. Сортировка содержимого одного или нескольких столбцов.
17. Арифметические операции над числовыми данными таблиц, использование формул. Построение диаграмм. Выбор типа диаграммы. Обрамление диаграммы рамкой. Изменение формата оси категорий и оси значений, формата оси сетки. Изменение названия диаграммы, оси значений, оси категорий.
18. Вставка рисунков. Способы связи рисунка с документом. Операции с рисунками: перемещение рисунка в другую позицию, масштабирование, изменение размера, создание рамки вокруг рисунка, редактирование. Использование графики в качестве подложки.
19. Понятие шаблона документа. Шаблоны документов, предоставляемые Word: типы, виды. Оформление документов с использованием готовых шаблонов. Создание собственных шаблонов на основе шаблона Обычный. Создание нового шаблона на основе текстового документа.
20. Списки простые и многоуровневые. Способы создания списков. Встроенные форматы списков. Изменение маркеров, создание произвольного символа маркера.

- Преобразование списков. Изменение положения маркера, положения текста относительно маркера страниц. Формат номера страницы.
21. Размещение в ячейках и использование данных разных типов в электронной таблице. Содержимое ячейки.
 22. Основные элементы электронной таблицы. Понятие адреса, ссылки. Типы ссылок. Вычисления по формулам и с помощью стандартных функций в электронных таблицах.
 23. Графическое представление табличных данных. Типы диаграмм. Понятия рядов данных и категорий значений. Элементы оформления диаграмм.
 24. Редактирование диаграмм. Форматирование типов диаграмм, элементов диаграммы, области диаграммы и области построения диаграммы, рядов данных, осей и сетки.
 25. Использование логических функций и функций для работы с базой данных в электронных таблицах. Сортировка данных в электронных таблицах.
 26. Понятие фильтрации данных в списке. Команда фильтрации, Автофильтр, расширенный фильтр.
 27. Информационный подход к анализу процессов и явлений. Основные стадии анализа. Пример информационного анализа.
 28. Качественные и количественные факторы. Оцифровка и нормализация значений факторов.

Вопросы для подготовки к промежуточной аттестации по дисциплине (экзамену)

1. Информатика как фундаментальная наука и область практической деятельности. Предмет научной информатики. Общекультурная роль информатики. Понятие об информационном обществе.
2. Информация как базовое понятие науки. Информационные системы и технологии. Этапы развития информационных технологий.
3. Фазы информационного процесса для получения информации. Этапы преобразования информации. Данные и знания.
4. Синтаксический аспект информации. Определение количества информации по Шеннону. Формулы кодирования и Хартли, примеры их применения.
5. Семантический и прагматический аспекты информации.
6. Представление дискретной информации в компьютере. Принципы сжатия. Примеры представления текстовой и графической информации.
7. Представление непрерывной информации в компьютере. Емкость канала передачи. Примеры представления звуковой информации.
8. Информационный подход к анализу процессов и явлений. Основные стадии анализа. Пример информационного анализа.
29. Качественные и количественные факторы. Оцифровка и нормализация значений факторов.
30. Понятие программного обеспечения. Классификация программных средств. Структура системного программного обеспечения. Операционные системы и их характеристики.
31. Сервисное программное обеспечение. Системные утилиты. Программы резервного копирования. Архиваторы и примеры их применения.
32. Вирусы и антивирусное программное обеспечение. Защита от сетевых атак.
33. Классификация программных средств для обработки текстовых документов.
34. Варианты создания документа в текстовом редакторе. Редактирование текстовых документов: цели и средства.

35. Форматирования текстовых документов: цели и средства. Вставка специальных элементов в текстовый документ (сноски, примечания, аннотации, специальные символы и др.).
36. Создание и редактирование таблиц в текстовом документе.
37. Вставка в текстовый документ оглавления, указателей, списка иллюстраций.
38. Операции с графическими объектами в текстовом редакторе.
39. Подготовка больших документов к печати (выделение разделов, нумерация страниц, формирование колонтитулов, предварительный просмотр).
40. Вычисления в текстовом процессоре; использование вычисляемых полей и формул.
41. Настройка окна табличного редактора, режимы просмотра документа. Навигация в текстовых документах с помощью закладок, переходов, гиперссылок.
42. Размещение в ячейках и использование данных разных типов в электронной таблице.
43. Цели и средства форматирования элементов электронных таблиц. Создание и применение пользовательских форматов.
44. Использование последовательностей и автозаполнения в электронных таблицах.
45. Вычисления по формулам и с помощью стандартных функций в электронных таблицах.
46. Графическое представление табличных данных. Типы диаграмм. Понятия рядов данных и категорий значений. Элементы оформления диаграмм.
47. Сортировка данных в электронных таблицах.
48. Работа с электронной таблицей как с базой данных.
49. Использование логических функций и функций для работы с базой данных в электронных таблицах.
50. Определения понятия задача. Фазы постановки задач.
51. Основные стадии компьютерного решения прикладной задачи.
52. Общее понятие алгоритма. Свойства компьютерных алгоритмов.
53. Теория алгоритмов.
54. Машина Тьюринга.
55. Теория вычислительной сложности.
56. Методика исследования вычислительной сложности задачи.
57. Способы представления алгоритмов.
58. Блок-схемы.
59. Основные алгоритмические конструкции: следование, цикл, переход, ветвление.
60. Стандартные алгоритмы.
61. Язык программирования как средство представления алгоритма.
62. Классификация языков программирования. Подходы к выбору языка. Версии языка.
63. Содержание и порядок выполнения отдельных этапов создания программы. Компоненты и возможности системы программирования. Модули системы.
64. Понятие типов данных в языке программирования.
65. Основные компоненты языка программирования: алфавит, словарь, набор операций, выражения, операторы, синтаксис, семантика.
66. Структура программы на языке Паскаль. Назначение и оформление раздела описания данных.
67. Описание одномерных и двумерных массивов. Размещение массивов в памяти.
68. Назначение процедур. Правила описания процедур и обращения к ним.
69. Присваивание значений переменным. Структура оператора присваивания. Понятие совместимости типов.
70. Операторы перехода и ветвления. Структура и синтаксис полного и неполного условного оператора.
71. Понятие заголовка и тела цикла. Варианты и синтаксис операторов цикла.
72. Применение, структура и синтаксис и вложенных циклов.

73. Работа с одномерными и двумерными массивами в циклах.
74. Варианты и синтаксис операторов ввода и вывода информации.

ГЛОССАРИЙ

Абзац - структурный элемент текста. В процессоре Word - произвольная последовательность символов между двумя символами "Возврат каретки" (CR), а также между началом текста и CR и концом текста.

Абсолютная ссылка — ссылка, которая в процессе копирования в составе формулы в другую ячейку не меняется

Абстрактные типы данных – теоретическая концепция, в которой пытаются полностью освободиться от явного задания представления данных и заменить его на порождение представления по свойствам операций, требуемым для нового типа данных.

Абстракция - важная характеристика сущности, отличающая ее от всех иных сущностей. Абстракция проводит границу между сущностями лишь с какой-то определенной точки зрения.

Автозамена - команда, используемая пользователем для замены названия элемента на сам элемент. Названия элементов должны быть уникальными сочетаниями символов, которые не могут встретиться в документе сами по себе.

Автомат - поведение, которое специфицирует последовательность состояний, через которые проходит объект на протяжении своего жизненного цикла, реагируя на события, включая описание реакций на эти события.

Автотекст - фрагмент документа, включающий текст или графику, который может использоваться для вставки в документ под управлением пользователя.

Агрегирование - специальный вид ассоциации, описывающий отношение между агрегатом (целым) и компонентом (частью).

Актер - множество логически связанных ролей, исполняемых при взаимодействии с прецедентами.

Активация - выполнение операции.

Активная область — часть таблицы, заполненная данными (только она хранится в файле)

Активный объект - объект, который владеет процессом или нитью и может инициировать управляющее воздействие.

Алгоритм - содержание и последовательность операций, четко определяющих решение задачи путем вычислительного процесса, преобразующего исходные данные в конечный результат

Алфавит - совокупность символов, отображаемых на устройствах печати и экранах и/или вводимых с клавиатуры терминала

Аргумент - фактическое значение, соответствующее формальному параметру.

Аргумент функции — величина, используемая для вычисления значения функции

Артефакт - элемент информации, используемый или порождаемый в процессе разработки программного обеспечения.

Архитектура - совокупность существенных решений об организации программной системы; набор структурных элементов и интерфейсов, из которых она состоит, вкупе с поведением, описываемым в терминах коопераций этих элементов; составление из данных структурных и поведенческих элементов все более крупных систем; архитектурный стиль, которому подчинена организация элементов, интерфейсов, коопераций и их композиции. К архитектуре программного обеспечения относятся не только структура и поведение, но также использование, функциональность, производительность, гибкость, повторное применение, ясность, экономические и технологические ограничения и компромиссы, а также эстетические аспекты.

Асинхронное действие - запрос, при котором посылающий объект не дожидается получения результата.

Атрибут - поименованное свойство классификатора, описывающее диапазон значений, которые могут принимать экземпляры этого свойства.

Атрибуты поля списка — характеристики типа данных поля; например, число с

двумя дробными разрядами

Бинарная ассоциация - ассоциация между двумя классами.

Блок (данных) — диапазон ячеек, содержащих данные, граничащий только с пустыми ячейками или границами окна документа

Блок-схема - графическое изображение алгоритма, подлежащего програм-мированию

Буквица - первая буква абзаца, занимающая несколько строк, которую “обтекает” остальной текст документа.

Булевский - перечислимый тип, значениями которого являются "истина" или "ложь".

Булевское выражение - выражение, результатом вычисления которого является булевское значение.

Ввод — операция внесения в память ПК информации (данных) для дальнейшей работы с ней

Версия - относительно полный и самосогласованный набор артефактов, предназначенный для внутреннего или внешнего использования.

Вешка разделения окна — элемент окна приложения Excel, с помощью которого любое окно рабочей книги м.б. разбито на две или четыре части с индивидуальной прокруткой в каждой

Взаимодействие - поведение, описываемое набором сообщений, которыми обмениваются между собой объекты в некотором контексте для достижения определенной цели.

Виджет – элемент дизайна веб-сайта или графический модуль, который служит для решения отдельных рабочих задач или быстрого получения информации.

Видимость - указывает, при каких обстоятельствах то или иное имя видимо и может быть использовано.

Вложенная функция — функция, возвращаемое значение которой является аргументом другой функции

Внедрение - четвертая фаза цикла разработки программного обеспечения, в течение которой оно передается пользователям.

Возвращаемое значение — величина, являющаяся результатом вычисления функции

Временное выражение - выражение, результатом вычисления которого является абсолютный или относительный момент времени.

Временный объект - объект, который существует только до тех пор, пока выполняется создавший его процесс или нить.

Выделение — процедура выбора ячейки или диапазона для последующей работы с ними, а также результат выполнения данной операции

Выравнивание — операция форматирования, с помощью которой достигается определенное расположение содержимого области отображения ячейки относительно ее границ

Выражение - строка, которая может быть использована для получения значения определенного типа.

Гарнитура - набор шрифтов определенного рисунка, но разных размеров (кеглей) и начертаний.

Гистограмма — тип графического изображения рядов данных в виде совокупности смежных по горизонтали прямоугольников, построенных на одной прямой, высоты которых пропорциональны определенным значениям рядов данных; их расположение на оси соответствует определенным категориям

Глобальная переменная - переменная, областью определения которой является вся программа или блок с вложенными в него подблоками

Граница — прямоугольная рамка, обозначающая выделенную ячейку или диапазон

График — линейная диаграмма с достаточно большой длиной ряда данных, обеспечивающей плавность кривой, которая отражает ту или иную закономерность (частный случай диаграммы)

Данные – информация, представленная в виде, пригодном для обработки автоматическими средствами при участии человека.

Данные динамической структуры (динамические структуры данных) – совокупность переменных, количество, размерность или характер взаимосвязей между которыми меняется во время работы программ. Динамические структуры данных базируются на указателях и динамических переменных.

Данные статической структуры – совокупность фиксированного количества переменных постоянной размерности с неизменным характером связей между ними.

Действие - выполнимое атомарное вычисление, которое приводит к изменению состояния системы или возврату значения.

Диаграмма - графическое представление множества элементов. Обычно изображается в виде графа с вершинами (сущностями) и ребрами (отношениями).

Диаграмма — графическое представление числовых данных в соответствии с некоторой системой условностей, определяемой типом диаграммы; графическое изображение соотношения каких-нибудь величин

Диаграмма деятельности - диаграмма, на которой представлены переходы потока управления от одной деятельности к другой. Диаграммы деятельности относятся к динамическому аспекту поведения системы. Это разновидность диаграмм состояний, где все или большая часть состояний являются состояниями деятельности, а все или большая часть переходов срабатывают при завершении деятельности в исходном состоянии.

Диаграмма с областями — график с заливкой пространства между линией графика и осью X

Диапазон ячеек — компактная группа ячеек таблицы, имеющая прямоугольную форму или совокупность таких компактных групп, не перекрывающих друг друга

Динамическая память (куча) – область памяти компьютера, рассматривая как массив байтов (например, в Turbo Pascal) и занимающая объем приблизительно 300000 байт, позволяющая обрабатывать такие структуры данных, как абстрактные типы данных.

Документ - объект обработки прикладной программы.

Дюйм - единица измерения длины. (1 дюйм равен 2,54 см.)

Заголовок поля списка — текстовая строка верхней ячейки столбца списка

Задача - путь выполнения программы, динамической модели или иного представления потока управления; процесс или нить.

Закладка - это имя, присвоенное некоторому месту в документе; она позволяет вам быстро перепрыгивать к этому месту или ссылаться на текст в этом месте с помощью перекрестных ссылок. Закладка может отмечать как курсор вставки, так и область выделения любого размера.

Замена - замена состоит в выделении ненужного места и вводе вместо него нового.

Запись – структура, связывающая элементы различных типов в один объект. Элементы в записи называются полями.

Запись списка — строка ячеек списка, соответствующая объекту или событию

Запрос - спецификация стимула, посылаемого объекту.

Зарезервированное слово - слово, смысл которого зафиксирован правилами языка и по которому транслятор распознает основные языковые конструкции

Значение - элемент области определения типа.

Имя - название сущности, отношения или диаграммы; строка, идентифицирующая элемент.

Инструментарий технологии программирования - это программные продукты, предназначенные для поддержки технологии программирования

Интервал между символами - интервал, определяющий расположение символов друг относительно друга. Различают обычный, разряженный, уплотненный интервал или установленный на заданную в пунктах величину.

Интернет (англ. inter + net - объединение сетей) – глобальная компьютерная сеть. В настоящее время Интернет можно рассматривать как одно из средств массовой

коммуникации наряду с прессой, радио и телевидением.

Интернет-технологии – общее название для телекоммуникационных технологий, построенных на основе сетевой архитектуры и протоколе обмена данными TCP/IP.

Интерфейс - множество операций, составляющее спецификацию услуг, которые предоставляет класс или компонент.

Интерфейс - совокупность средств и методов взаимодействия между элементами системы.

Интерфейс пользователя – разновидность интерфейсов, в котором одна сторона представлена человеком (пользователем), другая — машиной/устройством.

Инtranet – внутренняя сеть организации, доступ к ресурсам которой открыт только для членов организации.

Информационные технологии – общее название для технологий, концентрирующихся вокруг проблем обработки, хранения и управления данными.

Исполнение - прогон динамической модели.

Исследование - вторая фаза цикла разработки программного обеспечения, в ходе которой определяется общее видение продукта и его архитектура.

Итеративный подход: в контексте цикла разработки программного обеспечения - процесс управления потоком исполняемых версий.

Итерация - четко очерченный перечень работ, для которых определены конечная цель и критерий оценки. В результате нескольких итераций должна быть выпущена версия для внутреннего или внешнего использования.

Категория — сечение всех входящих в диаграмму рядов данных, отображаемое в соответствии названию или значению категории

Кегль - средний размер (высота) шрифта (полиграфический термин). Измеряется в пунктах.

Ключ сортировки — поле, по значению которого сортируется список

Колонтитул - структурный элемент документа. Находится в верхней или нижней части страницы и содержит некоторую информацию, идентифицирующую данный документ: номер страницы, номер раздела, название, дату, марку фирмы и т.п.

Кольцевая диаграмма — тип графического изображения одного или нескольких рядов данных, показывающего относительный вклад каждой точки данных в общий итог для конкретного ряда данных; каждый ряд отображается в собственном кольце; маркеры данных отображаются в виде сегментов кольца

Комментарий - аннотация, присоединенная к элементу или множеству элементов.

Компонент - программный модуль или объект, готовый для использования в качестве составного блока программы и которым можно визуальнo манипулировать во время разработки программы

Компонент - физическая заменяемая часть системы, реализующая спецификацию интерфейсов.

конкретной задачи.

Контейнер - объект, назначение которого - быть вместилищем других объектов; он предоставляет операции для доступа или последовательного обхода своего содержимого.

Контекст - множество взаимосвязанных элементов, предназначенное для определенной цели, например для специфицирования операции.

Контекстное меню — активизируемый нажатием правой кнопки мыши инструмент, обеспечивающий удобный и быстрый доступ к нужным командам; содержит только те команды, которые могут применяться к элементу, для которого активизировано это меню

Копилка - особая форма автотекста, которая работает так же, как буфер обмена, но с одним важным отличием. В копилке вы можете собирать текст и графику из различных частей документа, а затем вставлять их все вместе одновременно в новое место в документе.

Копирование ячеек — операция, посредством которой содержимое выделенной ячейки (диапазона) дублируется в другую ячейку (диапазон); при копировании область

приемника может быть больше области источника в целое число раз по любому из размеров

Кратность - спецификация диапазона возможных значений мощности множества.

Критерий выборки — логическое выражение, значение которого равно “истина” только для тех записей списка, которые интересуют нас в данный момент

Круговая диаграмма — тип графического изображения только одного ряда данных, показывающего относительный вклад каждой точки данных в общий итог для этого ряда данных.

Курсор - в текстовом режиме - мигающая полоска (метка) на экране, подчеркивающая позицию строки, в которую будет выведен символ при нажатии алфавитно-цифровой клавиши.

Легенда — совокупность условных обозначений и заголовков рядов данных, устанавливающих их однозначное соответствие

Лексика - совокупность правил образования цепочек символов (лексем), образующих идентификаторы (переменные и метки), операторы, операции и другие лексические компоненты языка.

Лепестковая диаграмма — тип графического изображения рядов данных, в котором значение каждой точки данных определяет ее расстояние от центральной точки; угол радиуса определяется положением точки в ряде данных.

Линейка - находится в верхней или левой части окна документа (горизонтальная или вертикальная линейка) Предназначена для контроля и обработки документа.

Линейный алгоритм - алгоритм, в котором действия осуществляются последовательно друг за другом

Линейчатая диаграмма — тип графического изображения рядов данных в виде совокупности смежных по вертикали прямоугольников, построенных на одной прямой, длины которых пропорциональны определенным значениям рядов данных; их расположение на оси соответствует определенным категориям

Лист рабочей книги — часть рабочей книги, имеющая имя

Логический алгоритм - алгоритм, в соответствии с которым решение поставленных задач сводится к логическим действиям

Маркер заполнения — маленький квадратик в правом нижнем углу границы активной ячейки (диапазона), с помощью которого в смежные (по горизонтали или вертикали) ячейки вводятся значения, образующие арифметическую прогрессию, либо копируются формулы

Маркеры — любой символ, который представляет точку данных на диаграмме; например, в гистограмме каждый столбец является маркером, в круговой диаграмме — каждый сектор, в графике или лепестковой диаграмме — небольшие квадраты, треугольники и другие символы, которые выводятся вдоль линий

Маркированный список - список, в котором каждый абзац отмечен каким-либо символом.

Мастер диаграмм — кнопка на стандартной панели инструментов, запускающая процесс создания внедренной диаграммы

Материнская строка — в структуре таблицы: расчетная строка, формулы которой составляются “вручную” и после копируются в другие расчетные строки

Машинно-ориентированный язык - язык программирования, отражающий структуру конкретного типа компьютера

Машинный язык - язык программирования, предназначенный для представления программ и данных в форме, пригодной для непосредственного восприятия их устройствами вычислительной машины

Межстрочный интервал - расстояние по вертикали между строками абзаца.

Моделирование — построение и изучение моделей; процесс изучения поведения модели в различных условиях с целью получения новых знаний о моделируемом объекте (процессе)

Модель - упрощение реальности, создаваемое для лучшего понимания разрабатываемой системы; семантически замкнутая абстракция системы.

Модель — совокупность объектов, ведущая себя в определенном аспекте аналогично изучаемому объекту или процессу; никогда не совпадает полностью с объектом (процессом)

Модульное программирование - способ разработки программ, при котором относительно независимые подзадачи программируются в виде отдельных программных модулей

Мощность множества - число элементов в множестве.

Начальная фаза - первая фаза цикла разработки программного обеспечения в которой исходная идея становится достаточно обоснованной, чтобы можно было принять решение о переходе к фазе исследования.

Начертание - текст в Windows может быть набран любым из четырех начертаний - обычным шрифтом, полужирным, курсивом и полужирным курсивом.

Неполнота - моделирование элемента, некоторые части которого отсутствуют.

Несовместимое подсостояние - подсостояние, в котором система не может находиться, одновременно находясь в других подсостояниях внутри одного и того же состояния.

Нисходящее программирование - способ разработки программ, при котором программирование ведется методом «сверху-вниз», от общего к деталям

Нумерованный список - список, в котором каждый абзац автоматически нумеруется (арабскими или римскими цифрами) или буквой алфавита.

Область действия - контекст, в котором употребление некоего имени является осмысленным.

Область отображения ячейки — структурная часть ячейки, содержимое которой отображается в таблице

Область содержимого ячейки — структурная часть ячейки, содержимое которой отображается в строке редактирования (формулы)

Объект - конкретная материализация абстракции; сущность с хорошо определенными границами, в которой инкапсулированы состояние и поведение; экземпляр класса.

Оператор – единица выполнения программы. Программа состоит из последовательности операторов. Выделяют два типа операторов: составной оператор (последовательность операторов, заключенная в фигурные скобки) и блок (совокупность описаний и операторов, заключенных в фигурные скобки).

Операция - реализация услуги, которая может быть запрошена у любого объекта класса.

Ориентация текста - в ячейках таблиц и надписях вы можете располагать текст как горизонтально, так и вертикально. В случае вертикальной ориентации у вас есть две возможности - сверху вниз или снизу вверх.

Особенность поведения - динамическая характеристика элемента, такого, как операция или метод.

Ось — вспомогательная линия на диаграмме, используемая для сопоставления графического изображения значения категории

Отбивка абзаца - в книгах и газетах между смежными абзацами иногда вставляют небольшой дополнительный промежуток (чтобы текст лучше читался). Этот промежуток называется отбивкой и может быть задан в полях Перед: (отбивка перед) и После: (отбивка снизу) окна << Абзац >>.

Отладка программы - этап разработки программы, состоящий в локализации, выявлении и устранении программных ошибок, факт существования которых уже установлен

Отладчик - программа, облегчающая программисту выполнение отладки разрабатываемых им программ

Относительная ссылка — это ссылка, которая в процессе копирования в составе формулы в другую ячейку автоматически изменяется; поведение относительных ссылок при переносе в составе формулы не меняется

Отступы - каждый абзац может иметь три вида отступов: слева, справа и по первой строке. С помощью отступов край абзаца можно сместить в центр страницы или утянуть его даже за край листа (что иногда называют “выступом”).

Оцифровка — совокупность надписей, отмечающих на осях опорные точки, соответствующие отдельным категориям или значениям

Панели инструментов — часть окна (либо самостоятельное окно), содержащая кнопки, нажатие которых приводит к выполнению тех или иных операций

Параметр - спецификация переменной, которая может быть изменена, передана или возвращена.

Параметризованный элемент - дескриптор элемента с одним или более несвязанными параметрами.

Перенос ячеек — операция, посредством которой содержимое выделенной ячейки (диапазона) перемещается в другую ячейку (диапазон); при переносе область источника и область приемника должны совпадать по размерам и форме

Перечислимый тип - список поименованных величин, образующих область значений некоторого атрибута.

Пиксель - минимальный элемент изображения ("точка") на экране монитора, создаваемый видеоадаптером. Имеет форму прямоугольника или квадрата.

Повторный ввод - специальная функция для повторного ввода текста (с помощью клавиши F4 или команды Вставка, Повторить ввод) как в месте вставки, так и поверх выделенного фрагмента.

Поле имени — часть строки редактирования формул, в которой отображается адрес (имя) активной ячейки (выделенного диапазона)

Поле списка — совокупность всех ячеек столбца списка, кроме первой

Поставщик - тип, класс или компонент, предоставляющий услуги, которые могут быть востребованы другими элементами.

Построение - третья фаза цикла разработки программного обеспечения, в ходе которой исполняемый архитектурный прототип доводится до состояния, когда он может быть передан пользователям.

Предметная область - область знаний или деятельности, характеризующаяся концепциями и терминами, понятными тем, кто работает в данной области.

Предусловие - ограничение, которое должно быть выполнено, когда вызывается операция.

Преобразование форматирования — приведение содержимого ячейки в соответствие с форматом, автоматически выполняющееся всякий раз при внесении изменений в таблицу

Примечание - графический символ для изображения ограничений или комментариев, присоединяемый к элементу или множеству элементов.

Примечание - некоторое пояснение к позиции текста, снабженное инициалами автора.

Примитивный тип - базовый тип, например "целое" или "строка".

Проблемно-ориентированный язык - язык программирования, предназначенный для решения задач определенного класса

Проверка орфографии - команда, выполняемая на базе основного словаря выбранного языка и дополнительного словаря пользователя, которые могут содержать произвольные слова (например, условные обозначения, ключевые слова языка программирования и т.д.).

Программа - последовательность указаний (команд или описаний и операторов), задающая алгоритм вычислительной машине

Программирование - процесс создания программы

Продукт - артефакт процесса разработки, такой как модель, код, документация и рабочий план.

Проекция - отображение множества на его подмножество.

Простые типы данных – целое, символьное, вещественное числа, число с плавающей точкой, типы перечисления, логический тип, указатель.

Процедура - поименованная часть программы, которая может выполнять некоторые четко заданные действия над условными данными, определяемыми с помощью формальных параметров

Процедурно-ориентированные языки - это языки, в которых имеется возможность описания программы как совокупности процедур или подпрограмм

Процесс - ресурсоемкий поток управления, который может выполняться параллельно с другими процессами.

Пункт - единица измерения размера (высоты шрифта, равная 1/72 дюйма).

Рабочая книга — объединенная под общим именем совокупность таблиц (листов), с помощью которых реализуются табличные модели; блокнот в 16 (по умолчанию) автономных листов-таблиц; основной документ Excel, соответствующий на диске файлу с расширением “.xls”; наименьшая единица хранения в файле

Рабочее пространство — часть окна приложения Excel, заключенная между строкой редактирования (формул) и строкой состояния

Разветвляющийся алгоритм - алгоритм, в котором действие выполняется по одной из возможных ветвей решения задачи, в зависимости от выполнения условий

Размер шрифта - средняя высота символа, измеряемая в пунктах.

Реализация (Realization) - семантическое отношение между

Редактирование — изменение значение ранее введенных данных

Редактирование диаграммы — изменение внешнего вида диаграммы, связанное с изменением объема данных, отображаемых на диаграмме, или типа диаграммы; добавление или удаление рядов (в том числе рядов значений, определяющих разметку осей); нанесение на диаграмму линий, стрелок, надписей, заголовков

Рекурсия – фундаментальное понятие в математике и компьютерных науках. В языках программирования рекурсивной программой называется программа, которая обращается сама к себе. Рекурсивная программа не может вызывать себя до бесконечности, следовательно, вторая важная особенность рекурсивной программы – наличие условия завершения, позволяющее программе прекратить вызывать себя.

Ряд данных — последовательность значений, размещенных в смежных ячейках электронной таблицы, задающая процесс, изменения которого необходимо представить графически

Сайт – в переводе с английского слово «сайт» означает место. Интернет-сайт является местом сосредоточения пользовательских файлов и информации, доступных через Интернет. Обычно, сайт имеет конкретный адрес или доменное имя.

свойствами ассоциации.

Свойство - поименованное значение, обозначающее некоторую характеристику элемента.

Семантика - смысловое содержание конструкций, предложений языка, семантический анализ – это проверка смысловой правильности конструкции

Семантическая ошибка - ошибка в программе, не связанная с нарушением синтаксиса языка программирования (неправильное описание алгоритма решения задачи, неверное определение типа или значения переменных, несогласованность исходных данных с алгоритмом, неверное использование правильных синтаксических конструкций языка)

Семантический анализ - выявление несоответствий типов и структур переменных, функций и процедур

Сервер интернет - компьютер, подключенный к сети, или выполняющаяся на нем программа, предоставляющие клиентам доступ к общим ресурсам и управляющие этими

ресурсами.

Сервисы сети Интернет – сервисы, которые могут быть предоставлены конечному пользователю при помощи Интернета. К ним относятся: электронная почта, передача данных, поиск информации и т.п.

Сигнатура - совокупность имени и параметров операции.

Синтаксис - совокупность правил образования языковых конструкций, или предложений языка программирования – блоков, процедур, составных операторов, условных операторов, операторов цикла и пр.

Синтаксическая конструкция - допустимое синтаксисом языка программирования сочетание символов или символов и зарезервированных слов

Синтаксическая ошибка - ошибка в программе, связанная с нарушением синтаксиса языка программирования (неправильное написание констант, переменных, выражений, операторов, процедур и других синтаксических конструкций)

Синтаксический анализ - проверка правильности конструкций, использованных программистом при подготовке текста

Система - множество элементов, организованных для достижения конкретной цели, разложенное на несколько подсистем и описываемое набором моделей, возможно с различных точек зрения.

Система программирования - совокупность средств разработки программ (языки программирования, текстовые редакторы, трансляторы, редакторы связей, библиотеки подпрограмм, утилиты и обслуживающие программы), обеспечивающих автоматизацию составления и отладки программ, подготовку соответствующей документации

Скрипт - программа, содержащая набор инструкций для некоторых приложений или утилит.

Смешанная диаграмма — совокупность двух и более диаграмм, одинакового или различных типов, построенных на одной оси категорий; в них может быть несколько осей значений

Сноска - структурный элемент текста. Примечание к тексту, которое находится в нижней части страницы или в конце документа и снабжается номером или другой пометкой.

Событие - спецификация существенного факта, имеющего положение в пространстве и во времени. В контексте автоматов событие - это возникновение стимула, который может активизировать переход из одного состояния в другое.

Сообщение - спецификация передачи информации между объектами в расчете на то, что за этим последует некоторая деятельность; прием сообщения обычно трактуется как возникновение события.

Сортировка – алгоритмический процесс перестановки объектов данной последовательности в определенном заданном порядке. Цель сортировки – облегчение поиска элементов в отсортированной последовательности.

Сортировка включениями (вставками) – метод сортировки, в котором из неупорядоченной последовательности поочередно выбирается каждый элемент, сравнивается с предыдущим (уже упорядоченным) элементом последовательности и помещается на соответствующее место в последнем.

Сортировка выбором – метод сортировки, при котором в неупорядоченной последовательности выбирается минимальный элемент, который исключается из дальнейшей обработки, а оставшаяся последовательность принимается за исходную. Процесс повторяется до тех пор, пока все элементы не будут выбраны.

Сортировка обменом (метод пузырька) – метод сортировки, при котором все соседние элементы массива попарно сравниваются друг с другом и меняются местами в том случае, если предшествующий элемент больше последующего.

Сортировка списка — упорядочение записей списка по возрастанию или убыванию значений одного или нескольких полей

Составные статические структуры данных – массивы, записи, структуры, файлы,

абстрактные типы данных и классы, стеки и очереди, параметризованные типы данных, списки, деревья, классы.

Состояние - ситуация в жизненном цикле объекта, во время которой он удовлетворяет некоторому условию, выполняет определенную деятельность или ожидает какого-то события.

Спецификация - текстовое объявление синтаксиса и семантики некоторого строительного блока; декларативное описание того, чем является или что делает некая сущность.

Списки-перечисления - несколько последовательных абзацев вы можете объявить списком-перечислением.

Список — прямоугольный массив смежных ячеек, удовлетворяющий следующим требованиям: 1) верхняя строка списка содержит текстовые заголовки каждого из столбцов списка; 2) остальные строки списка соответствуют каждая определенному событию или объекту; 3) каждый столбец содержит во всех ячейках данные одного типа; 4) пустые строки отсутствуют

Ссылка — часть формулы, указывающая, где располагаются данные, которые необходимо использовать вместо ссылки при вычислении формулы

Стартап - недавно образованная компания, которая начинает свой бизнес на основе инновационного развития, предлагая новые виды товаров и услуг, преимущественно в сфере информационных технологий или других наукоёмких отраслей промышленности.

Стиль - способ форматирования структурного элемента текстового документа (последовательности символов и абзаца).

Строка - последовательность символов, имеющих графическое начертание.

Строка состояния — часть окна приложения Excel, предназначенная для вывода сведений о текущем состоянии рабочей области, программы и текущих подсказок

Строка формул — часть окна приложения Excel, предназначенная для отображения и редактирования формул области содержимого активной ячейки

Строки – фундаментальный тип данных, предназначенный для ввода, обработки и вывода слов в алфавитах.

Структура данных - способ объединения, взаимосвязь или взаимное расположение нескольких элементов данных, рассматриваемых как одно целое

Структура данных – совокупность элементов данных, между которыми существуют некоторые отношения, причем элементами данных могут быть простые данные и структуры данных.

Структура данных – способ организации данных, применяющийся для решения задачи.

Структурное программирование - методология программирования, направленная на создание логически простых и понятных программ. Структурное программирование основано на предположении, что логичность и понятность программы облегчает разработку, доказательство правильности и последующее сопровождение программы, а также обеспечивает ее надежность

Схема документа - режим схемы документа показывает различные заголовки и соответствующие им части текста. Это очень похоже на режим электронного документа, но Word организует схему автоматически. Вы можете использовать схему документа для быстрого перехода к нужным частям текста без утомительного постраничного перелистывания.

Текстовый файл - файл, который содержит совокупность строк переменной длины (чаще от 0 до 255) , причем каждая строка - это совокупность произвольных символов кодовой таблицы, замкнутая двумя символами с кодами 13 (возврат каретки) и 10 (новая строка).

Текущая строка — строка, в которой располагается активная ячейка

Тестирование программы - процесс выполнения программы с целью обнаружения ошибок путем реального выполнения специально подобранных контрольных примеров

Технология программирования - технология разработки программного средства как совокупность абсолютно всех технологических процессов его создания – от момента зарождения идеи о данном программном средстве до составления необходимой программной документации

Тип данных – множество значений, набор операций, которые можно применять к таким значениям, и, возможно, способ реализации хранения значений и выполнения операций.

Тип диаграммы — система условностей, определяющих связь исходных данных с изображением диаграммы

Точечная (фазовая) диаграмма — тип графического изображения рядов данных, в котором местоположение точки определяется значениями двух рядов данных, одно из которых задает ее горизонтальную, а второе — вертикальную координату; отсутствует ось категорий и обе оси декартовых координат являются осями значений; обычно используется для определения типа зависимости между двумя рядами данных.

Трассировка программы - выполнение программы или ее участка, сопровождающееся выводом на экран, принтер или другой регистрацией в хронологической последовательности информации о событиях, связанных с выполнением программы

Требование - желаемая функциональность, свойство или поведение системы.

Узел - физический элемент, существующий во время выполнения системы и представляющий вычислительный ресурс, который обладает, по меньшей мере памятью, а зачастую также и процессором.

Управляемый прецедентами: в контексте цикла разработки программного обеспечения - процесс, в котором прецеденты служат основным артефактом для формулирования желаемого поведения системы, для верификации и контроля системной архитектуры, для тестирования и для обмена информацией между участниками проекта.

Управляемый рисками: в контексте цикла разработки программного обеспечения - процесс, в котором при выпуске каждой новой версии основное внимание обращается на выявление и уменьшение факторов, представляющих наибольший риск для успешного завершения проекта.

Уровень абстракции - точка в иерархии абстракций, нисходящей от верхних (очень абстрактных) до нижних (очень конкретных) уровней.

Фаза - промежуток времени между двумя опорными точками в процессе разработки, в течение которого должны быть достигнуты заранее поставленные хорошо определенные цели, артефакты доведены до готовности и принято решение о том, следует ли переходить к следующей фазе.

Фактический параметр - аргумент функции или процедуры.

Фильтрация списка — сокрытие (но не удаление) в списке всех строк, не отвечающих условию отбора

Фон — определенное цветовое оформление области отображения ячейки

Фоновая печать - это возможность продолжения работы в Word во время печати документа.

Фоновое сохранение - когда вы сохраняете документ, Word производит запись на диск в фоновом режиме. Это означает, что вы можете продолжать свою работу – Word внесет все изменения на диск, не отрывая вас от дела.

Форма данных — диалоговое окно, в котором полям списка соответствуют поля ввода и имеются кнопки для выполнения операций со списком

Формат — описание способа отображения данных на экран или при печати документа

Форматирование — изменение представления ранее введенных данных

Форматирование диаграммы — изменение ее внешнего вида в пределах заданного типа диаграммы, не связанное с изменением объема данных, отображаемых на диаграмме; включает: форматирование ряда данных, оси, размера и легенды

Формула — находящаяся в области содержимого ячейки строка, начинающаяся со знака “=” и описывающая действия, выполнение которых приводит к вычислению значения, помещаемого в область отображения той же ячейки; программа

Функция — часть формулы, указывающая на необходимость выполнения некоторых стандартных вычислений; состоит из имени функции и следующего за ним в скобках через точку с запятой списка аргументов

Функция в программе - поименованная часть программы, результатом выполнения которой является значение, присваиваемое имени функции

Хостинг – услуга размещения клиентских файлов на интернет-сервере. Тем самым достигается доступ к этим файлам через Интернет. Различают коммерческий и бесплатный хостинги. Тарифы на услуги хостинга сильно различаются в зависимости от набора оказываемых услуг: регистрации доменных имён, дискового пространства, установленного программного обеспечения, баз данных и т.п.

Цикл с постусловием - цикл, у которого тело расположено перед проверкой условия

Цикл с предусловием - цикл, у которого тело расположено после проверки условий

Циклическая ссылка — формула, возвращаемое значение которой зависит от собственного значения

Циклический алгоритм - алгоритм, в котором некоторая часть операций (тело цикла – последовательность команд) выполняется многократно

Численный алгоритм - алгоритм, в соответствии с которым решение поставленных задач сводится к арифметическим действиям

Числовой шаблон — строка специальных символов, задающая, каким образом число выводится на экран; стандартные шаблоны содержатся в списке “Числовые форматы”

Шаблон - это образец, модель или схема построения нового объекта. В MS Word шаблоны используются для создания документов.

Экземпляр- конкретная материализация абстракции. К этой сущности могут быть применены операции; она обладает состоянием, в котором запоминаются результаты операций.

Электронная коммерция – любые формы торговых сделок, при которых взаимодействие сторон осуществляется с применением возможностей информационных и телекоммуникационных технологий

Электронная почта – интернет-сервис, позволяющий отправлять, получать и хранить электронные сообщения. Также позволяет пересылать файлы. Основной инструмент официального или делового общения в Интернете.

Электронный бизнес - процессы внедрения и использования новых информационных технологий, вычислительной техники, телекоммуникационных сетей (включая Интернет) для достижения бизнес-задач.

Элемент - атомарная составляющая модели.

Эффект шрифта - это образец, модель или схема построения нового объекта. В MS Word шаблоны используются для создания документов.

Юзабилити – удобство использования сайта для его посетителей, логичность и простота в расположении элементов управления.

Язык программирования - это формализованный язык для описания алгоритма решения задач на компьютере

Ярлык листа — элемент окна приложения Excel, содержащий имя листа рабочей книги

Ячейка таблицы — основная структурная единица электронной таблицы, представляющая собой имеющую собственное имя (адрес) прямоугольную область рабочего пространства приложения; состоит из области содержимого и области отображения.

МЕТОДИЧЕСКИЕ РЕКОМЕНДАЦИИ ДЛЯ ПРЕПОДАВАТЕЛЯ ПО ДИСЦИПЛИНЕ

Основной целью изучения дисциплины «Информатика и программирование» является освоение студентами углубленных основ информатики и современных информационных технологий, совершенствование навыков работы на компьютере и использование этих навыков в практической деятельности. Дисциплина изучает углубленные основы информатики, программирования и современных информационных технологий, основ представления знаний с помощью информационных технологий.

Форма промежуточной аттестации знаний — зачет, экзамен.

Лекционное занятие — это систематическое, последовательное, устное изложение лектором учебного материала. Занятие «лекция» носит, прежде всего, обзорный характер, охватывая весь круг выносимых на изучение учебных вопросов. При проведении такого типа занятий очень важно живое слово лектора, его педагогическое мастерство как педагога, который дает студентам информационную базу. Лекции являются важной формой передачи преподавателем студентам общетеоретических знаний.

Лекции, как правило, читаются не по всем, а по наиболее сложным темам курса, не дублируют учебники, а содержат новейшие научные данные и примеры, которых может не быть в учебных пособиях. Для лучшего усвоения материала на лекционных занятиях целесообразно предварительно перед лекцией ознакомиться с положениями лекционной темы в конспекте лекций, содержащемся в данном учебно-методическом пособии либо в рекомендуемых учебниках.

Практическое (подгрупповое) занятие — другая важная форма учебного процесса. Они способствуют закреплению и углублению знаний, полученных студентами на лекциях и в результате самостоятельной работы над научной и учебной литературой и нормативными источниками. Они призваны развивать самостоятельность мышления, умение делать выводы, связывать теоретические положения с практикой, формировать профессиональное правовое сознание. На занятиях вырабатываются необходимые каждому бакалавру навыки и умения публично выступать, логика доказывания, культура профессиональной речи. Кроме того, занятия — это средство контроля преподавателей за самостоятельной работой студентов, они непосредственно влияют на уровень подготовки к итоговым формам отчетности — зачетам и экзаменам. В работе и выступлении на занятии должны содержаться следующие элементы:

- четкое формулирование соответствующего теоретического положения в виде развернутого определения;
- приведение и раскрытие основных черт, признаков, значения и роли изучаемого явления или доказательства определенного теоретического положения;
- подкрепление теоретических положений конкретными фактами.

Для качественного и эффективного изучения актуальных проблем теории необходимо овладение навыками работы с книгой, воспитание в себе стремления и привычки получать новые знания из научной и иной специальной литературы. Без этих качеств не может быть настоящего специалиста ни в одной области деятельности.

Читать и изучать, следует, прежде всего, то, что рекомендуется к каждой теме программой, планом семинарских занятий, перечнем рекомендуемой литературы.

Когда студент приступает к самостоятельной работе, то он должен проявить инициативу в поиске специальных источников. Надо иметь в виду, что в каждом последнем номере издаваемых журналов публикуется библиография всех статей, напечатанных за год, это облегчает поиск нужных научных публикаций.

Работа с научной литературой, в конечном счете, должна привести к выработке у бакалавра умения самостоятельно размышлять о предмете и объекте изучения, которое должно проявляться:

- в ясном и отчетливом понимании основных понятий и суждений, содержащихся в публикации, разработке доказательств, подтверждающих истинность тех или иных положений;
- в понимании студентами обоснованности и целесообразности, приводимых в книге и статье примеров, поясняющих доказательства и выводы автора. При этом будет уместно, если бакалавр самостоятельно приведет дополнительные примеры к этим выводам;
- в отделении основных положений от дополнительных, второстепенных сведений;
- в способности студента критически разобраться в содержании публикации, определить свое отношение к ней в целом, дать ей общую оценку, характеристику.

Другим важнейшим методическим приемом в учебном процессе является самостоятельная работа студента.

Самостоятельная работа в высшем учебном заведении, является важной организационной формой индивидуального изучения студентами программного материала.

В современных условиях дидактическое значение самостоятельной подготовки неизмеримо возрастает, а ее цели состоят в том, чтобы:

- повысить ответственность самих обучаемых за свою профессиональную подготовку, сформировать в себе личностные и профессионально-деловые качества;
- научить студентов самостоятельно приобретать знания, формировать навыки и умения, необходимы для юридической деятельности;
- развивать в себе самостоятельность в организации, планировании и выполнении заданий, определяемых учебным планом и указаниями преподавателя.

Достигнуть этих целей в ходе самостоятельной работы при изучении дисциплины возможно только при хорошей личной организации своего учебного труда, умении использовать все резервы имеющегося времени и подчинить их профессиональной подготовке.

Самостоятельная работа как метод обучения включает:

- изучение и конспектирование обязательной литературы в соответствии с программой дисциплины;
- ознакомление с литературой, рекомендованной в качестве дополнительной;
- изучение и осмысление специальной терминологии и понятий;
- изучение и отработка нормативных актов, комментариев к ним, проведение сравнительного анализа с предыдущим;
- сбор материала и написание контрольных, конкурсных и дипломных работ;
- изучение указанной литературы для подготовки к зачету.
- Основными компонентами содержания данного вида работы являются:
- творческое изучение учебных пособий и научной литературы;
- умелое конспектирование;
- участие в различных формах учебного процесса, научных конференциях, в работе кружков и т. Д.;
- получение консультаций у преподавателя по отдельным проблемам курса;
- получение информации и опыта о работе профессионалов в процессе производственно-учебной практики;
- знакомство со специальной литературой при формировании своей личной библиотеки и др.

Данный комплекс рекомендаций позволяет студентам овладеть многими важными приемами самостоятельной работы и успешно использовать их при подготовке контрольных по дисциплине.

Важнейшей формой учебной отчетности студента являются индивидуальные задания по практическим работам. Выполнение индивидуальных заданий по практическим

работам является промежуточной формой отчетности по изучаемой дисциплине и преследует цель лишь оценить способность студента к самостоятельному поиску источников, формированию содержания и его письменного изложения по указанной проблеме. Это важная составляющая изучения дисциплины, а также эффективная форма контроля знаний.

Как правило, индивидуальных заданий по практическим работам по дисциплине сугубо индивидуальны, то есть их тематика персонифицирована. Однако в отдельных случаях темы работ могут быть адресованы и сразу нескольким бакалаврам, и группе в целом. Таким приемом преподаватель выявляет степень усвоения какой-то важной учебной проблемы и определяет необходимость проведения дополнительных занятий по какой-либо теме.

Игра позволяет влиять на правовые установки студентов. Учебно-правовые ситуации относятся к тем методическим средствам, которые позволяют осуществлять взаимосвязь понятийно-категориального уровня правосознания с поведенческим. В результате достигается не только интеллектуальный, но и эмоциональный уровень усвоения правовых понятий и идей.

Учебно-тренировочные ситуации являются специфическим методическим приемом, одним из основных видов проблемно-развивающего обучения, благодаря которому усиливается практический интерес бакалавров к теоретико-правовым вопросам.

Эффективность применения учебных ситуаций зависит от соблюдения следующих условий: знание студентами теоретического материала и наличие достаточного личного опыта и жизненного опыта вообще.

Важными в методическом плане на семинарских занятиях являются проводимые **тестовые опросы** и решение задач, которые содействуют превращению знаний в глубокие убеждения, дают простор для развития творческо-эмоциональной сферы, позволяют сделать выводы об эффективности занятий с учащимися, что в итоге повышает интерес к овладению знаниями.

Только сочетая дидактически и органически все методические способы и приемы в их диалектическом единстве и взаимосвязи мы можем добиться должного уяснения учебного материала со стороны студентов.

Тема занятия	Виды учебных занятий	Способы учебной деятельности	Методы обучения, формы педагогического общения	Средства обучения	Формы контроля
Информация и информационные процессы	Лекция, подгрупповое занятие	Коллективный	Методы: объяснительно-иллюстративный, репродуктивный. Формы: монолог/диалог	Проектор, компьютеры с установленным программным обеспечением, презентация, электронный курс по дисциплине	Устный опрос Проверка файлов общих и индивидуальных заданий, устный опрос, оперативное исправление ошибок
Программные средства обработки информации	Лекция, подгрупповое занятие	Коллективный	Методы: объяснительно-иллюстративный, репродуктивный.	Проектор, компьютеры с установленным	Устный опрос Проверка файлов

			Формы: монолог/диалог	программным обеспечением, презентация, электронный курс по дисциплине	общих и индивидуальных заданий, устный опрос, оперативное исправление ошибок
Постановка и алгоритмизация прикладных задач	Лекция, подгрупповое занятие	Индивидуально-групповой	Методы: объяснительно-иллюстративный, репродуктивный. Формы: монолог/диалог	Проектор, компьютеры с установленным программным обеспечением, презентация, электронный курс по дисциплине	Проверка индивидуальных заданий
Методы и языки программирования	Лекция, подгрупповое занятие	Индивидуально-групповой	Методы: объяснительно-иллюстративный, репродуктивный. Формы: монолог/диалог	Проектор, компьютеры с установленным программным обеспечением, презентация, электронный курс по дисциплине	Проверка индивидуальных заданий

Тематический план изучения дисциплины «Информатика и программирование»

Год набора с 2019 форма обучения очная

Наименование разделов и тем	Всего	Трудоемкость по дисциплине				СРС	Формируемые компетенции
		контакт т. работа	в т.ч.				
			лекции и	лаб. работы	практ. / сем. \ИЗ		
Информация и информационные процессы	68	18	8	10		50	УК-1, ОПК-1
Программные средства обработки информации	65	18	10	8		47	ОПК-1, ОПК-4
Постановка и алгоритмизация прикладных задач	76	26	8	18		50	ОПК-4, ОПК-7
Методы и языки программирования	88	28	10	18		60	ОПК-4, ОПК-7
Экзамен							
Зачет	27	27					
Итого по дисциплине	324	117	36	54		207	
Зачетных единиц	9						
Курсовая работа	+						

Тематический план изучения дисциплины «Информатика и программирование»

Год набора с 2020 форма обучения заочная

Наименование разделов и тем	Всего	Трудоемкость по дисциплине				СРС	Формируемые компетенции
		контакт. работа	в т.ч.				
			лекции и	лаб. работы	практ./ сем. \ИЗ		
Информация и информационные процессы	80	6	2	4		74	УК-1, ОПК-1
Программные средства обработки информации	81	10	4	6		71	ОПК-1, ОПК-4
Постановка и алгоритмизация прикладных задач	74	4	2	2		70	ОПК-4, ОПК-7
Методы и языки программирования	76	6	2	4		70	ОПК-4, ОПК-7
Контроль	13	13					
Итого по дисциплине	324	39	10	16		285	
Зачетных единиц	9						
Курсовая работа	+						
Контрольная работа	+						