

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ
КЫРГЫЗСКОЙ РЕСПУБЛИКИ
КЫРГЫЗСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
им. И. РАЗЗАКОВА
БИШКЕКСКИЙ ТЕХНИЧЕСКИЙ КОЛЛЕДЖ

«Согласовано»

Методист



« 01 » сентября 2023 год



«Утверждаю»

Зам. Директора по УР



Долотов М.М.

« ___ » _____ 2023 год

Рабочая программа

По предмету: Технология разработки программных продуктов

Для специальности: 230110 «Техническое обслуживание средств вычислительной техники и компьютерных сетей»

Курс III семестр 5

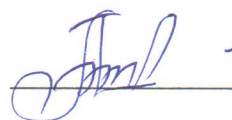
Количество часов по учебному плану: 180 часа

Разработана преподавателем БТК КГТУ Уркунбаева А.К

Составлен на основании ГОС СПО КР по специальности 230109 «Программное обеспечение вычислительной техники и автоматизированных систем», утвержденного приказом МОиН КР №567/1 от 15.05.2019 года (регистрационный №180 от 07.08.2019 МЮ КР)

Рассмотрено и одобрено на заседании ПЦК СД: Батырбекова Д.

Протокол № 4 от «30» 08 2023 года.



Бишкек 2023

1. Паспорт рабочей программы учебной дисциплины «Технология разработки программного продукта»

1.1 Область применения программы

Рабочая программа учебной дисциплины «Технология разработки программного продукта» является частью основной профессиональной образовательной программы в соответствии с ГОС СПО по специальности 230109 «Программное обеспечение вычислительной техники и автоматизированных систем», утвержденный приказом Министерства образования и науки КР №567/1 от 15.05.2019 года (регистрационный № 180 от 07.06.2019 года МЮ КР)

Изучение «Технологии разработки программного продукта» направлено на достижение следующих *целей*:

1.2. Цели учебной дисциплины

Целью данного курса является изучение среды программирования Java, которая является средой разработки, используемой для поддержки и разработки приложений, предназначенных как для рабочих станций, так и для серверов.

1.3. Место дисциплины в структуре основной профессиональной образовательной программы:

Этот курс является составной частью профессионального цикла, определяющих подготовку студентов в области современных информационных технологий.

Задачи учебной дисциплины: для достижения цели студент должен

знать:

- основные этапы технологии проектирования программных продуктов;
- приемы оптимизации программ;
- особенности модульного программирования;
- инструментальные средства разработки программ;
- принципы и методы коллективной разработки программных средств;

уметь:

- разрабатывать алгоритм программной реализации поставленной задачи;
- создавать программный продукт по разработанному алгоритму;
- выполнять отладку и тестирование программного продукта;

владеть:

- навыками выбора, проектирования и реализации алгоритмов для решения профессиональных задач;
- современными технологиями и средствами проектирования, тестирования программных приложений.

1.4 Перечень формируемых компетенций

При освоении основной профессиональной образовательной программы студента должен овладеть следующими компетенциями

Общие:

- ОК 1.** Уметь организовать собственную деятельность, выбирать методы и способы выполнения задач, оценивать их эффективности и качество;
- ОК 2.** Решать проблемы, принимать решения стандартных и нестандартных ситуациях, проявлять инициативу и ответственность;
- ОК 3.** Осуществлять поиск, интерпретацию и использование информации, необходимой для эффективного выполнения профессиональных задач, профессионального и личностного развития;
- ОК 4.** Использовать информационно-коммуникационные технологии в профессиональной деятельности;
- ОК 5.** Уметь работать в команде, эффективно общаться с коллегами, руководством и клиентами;
- ОК 6.** Брать ответственность за работу членов команды(подчиненных) и их обучение на рабочем месте, за результат выполнение задания;
- ОК 7.** Управлять собственным личностным и профессиональным развитием, адаптироваться к изменениям условий труда и технологий в профессиональный деятельности;
- ОК 8.** Быть готовым к организационно-управленческой работе с малыми коллективами;
- ОК 9.** Быть способным анализировать и оценивать социально- экономические и культурные последствия новых явлений в науке, технике и технологии, профессиональной сфере;
- ОК 10.** Быть способным на научной основе оценивать свой труд; оценивать большой степенью самостоятельности результаты своей деятельности;

Профессиональные:

- ПК 1.** Выполнять разработку спецификацией отдельных компонентов;
- ПК 2.** Осуществлять разработку кода программного продукта на основе готовых спецификаций на уровне модуля;
- ПК 3.** Выполнять отладку программных модулей с использованием специализированных программных средств;
- ПК 4.** Выполнять тестирование программных модулей;
- ПК 5.** Осуществлять оптимизацию программного кода модуля;
- ПК 6.** Разрабатывать компоненты проектной и технической документации с использованием графических языков спецификации.

1.5 Рекомендуемое количество часов на основе рабочей программы учебной дисциплины:

Максимальная учебная нагрузка студентов 180 часов в том числе:

- Обязательная аудиторная учебная нагрузка -108 часов;
- Самостоятельная работа студентов – 72 часов;
- Курсовая работа -18 часов;

2. Структура и содержание учебной дисциплины

2.1. Объем учебной дисциплины и виды учебной работы

Вид учебной работы	Объем часов
Максимальная учебная нагрузка (всего)	180
Обязательная аудиторная нагрузка (всего)	108
В том числе:	
Теоретические занятия	30
Практические занятия	60
Курсовая работа	18
Самостоятельная работа студента	72
Итоговая аттестация в форме экзамена	

2.2. Примерный тематический план и содержание учебной дисциплины

№	Наименование разделов и тем	Ауд. 108 ч	СРС 72 ч	В том числе:			Уровень освоения
				Тео. 30	ЛПЗ 60	КП 18	
1	2	3		4	5	6	7
Раздел I. Общие принципы разработки программных продуктов							
1.1	Введение. Программные продукты и их основные характеристики	2		2			2
1.2	Классификация программных продуктов	4		2	2		2
1.3	Особенности создания программного продукта	4	2	2	2		2
1.4	Жизненный цикл программы	4	2	2	2		2
1.5	Проектирование программных продуктов	4	2	2	2		2
1.6	Структура и формат статические и динамические данные	2	2	2			2
	Итого по разделу:	20	8	12	8		
Раздел 2. Методология проектирования программных продуктов							
2.1	Методы проектирования ПП	4	2	2	2		2
2.2	Структура ПП	4	2	2	2		2
2.3	Проектирование интерфейса пользователя	4	2	2	2		2
	Итого по разделу:	12	6	6	6		
Раздел 3. Разработка программных продуктов							
3.1	Стиль программирования	4	2	2	2		2
3.2	Языки программирования	2	2		2		2
3.3	Модульное программирование	2	2		2		2
3.4	Структурное программирование	4	2	2	2		2
3.5	Объектно-ориентированное программирование	4	2		4		2

3.6	Эффективность и оптимизация программ	4	2	2	2		2
3.7	Обеспечение качества программного продукта	4	2	2	2		2
	Итого по разделу:	24	14	8	16		
Раздел 4. Отладка, тестирование и сопровождение программ							
4.1	Ошибки программного обеспечения	4	4		4		2
4.2	Отладка программ	4	4		4		2
4.3	Тестирование программ	4	6		4		2
4.4	Сопровождение программ	4	2		4		2 ^a
4.5	Защита программ	4	2		4		2
4.6	Пакеты прикладных программ	6			6		2
	Итого по разделу:	26	18		26		
Раздел 5. Инструментальные средства разработки программ							
5.1	Общая характеристика инструментальных средств разработки программ	2	2		4		2
5.2	Применение CASE-средств	2	2		2		2
	Итого по разделу:	4	4		6		
Раздел 6. Коллективная разработка программных средств							
6.1	Организация работ при коллективной разработке программных продуктов	2		2			2
6.2	Экономические аспекты создания программных средств	2		2			2
	Итого по разделу:	4		4			
Раздел 7. Курсовая работа							
7.1	Курсовая работа	18	18			18	18
	Итого по разделу:	18	18			18	
	Итого по предмету	108	72	30	60	18	

Для характеристики уровня освоения учебного материала используются следующие обозначения:

1. Ознакомительный (узнавание ранее изученных объектов, свойств);
2. Репродуктивный (выполнение деятельности по образцу инструкции или под руководством);
3. Продуктивный (планирование и самостоятельное выполнение деятельности решение проблемных задач).

СОДЕРЖАНИЕ ДИСЦИПЛИНЫ

Раздел I. Введение. Общие принципы разработки программных продуктов

Тема 1.1 Программные продукты и их основные характеристики (2 ч)

Основные понятия программного обеспечения. Программа, программное обеспечение, задачи и приложения. Технологические и функциональные задачи. Процесс создания программ: постановка задачи, алгоритмизация, программирование. Понятие

программного продукта. Характеристика программного продукта и его специфика. Показатели качества программного продукта: мобильность, надежность, эффективность, легкость применения, модифицируемость, коммуникативность.

Тема 1.2 Классификация программных продуктов (2 ч)

Классификация программных продуктов по сфере использования: системное программное обеспечение, инструментарий технологий программирования, пакеты прикладных программ. Состав и назначение инструментария технологий программирования. Средства для создания приложений. CASE-технологий. Программные продукты для создания приложений. Пакеты прикладных программ. Характеристика классов пакетов прикладных программ.

Практическая работа №1 (2 ч)

1. Основные понятия программного обеспечения
2. Классификация программных продуктов
3. Системное программное обеспечение
4. Пакеты прикладных программ
5. Инструментарий технологии программирования

1.3. Особенности создания программного продукта (2 ч)

Технология программирования. Программа. Программное обеспечение (software). Приложение (application). Постановка задачи (problem definition). Алгоритм.

Практическая работа №2 (2 ч)

Управление требованиями

- постановка задачи
- выбор критериев эффективности
- проведение предварительных научно-исследовательских работ;
- разработка ТЗ.

Эскизный проект:

- структура входных и выходных данных;
- уточнение методов решения; общий алгоритм;
- разработка документации эскизного проекта.

Технический проект:

- уточнение структуры входных и выходных данных;
- разработка алгоритмов;
- формы данных;
- семантика и синтаксис языка;
- структура программы; конфигурация технических средств;
- план работ.

Рабочий проект:

- программирование и отладка;
- разработка документов;
- подготовка и проведение испытаний;
- корректировка программы и документов по итогам испытаний.

Внедрение:

- передача программы и документов для сопровождения;
- оформление акта;
- передача в Фонд алгоритмов и программ

Тема 1.4 Жизненный цикл программ (2 ч)

Понятие жизненного цикла программы и его этапы. Анализ требований к программе, определения спецификации программы, проектирование, кодирование и тестирование, эксплуатация и сопровождения программы. Характеристики этапов жизненного цикла программного продукта и его специфика. Особенности разработки программного продукта.

Практическая работа №3 (2 ч)

1. Жизненный цикл разработки ПО
2. Сбор и анализ требований (Planning and Requirement Analysis)
3. Документирование требований (Defining Requirements)
4. Дизайн (Design the Product Architecture)
5. Разработка ПО (Building or Developing the Product)
6. Тестирование (Testing the Product)
7. Внедрение и поддержка продукта (Deployment in the Market and Maintenance)

Тема 1.5 Проектирование программных продуктов (2 ч)

Технологический процесс разработки программного обеспечения. Стадии разработки программ и программной документации. Сопровождаемая документация. Основные требования к содержанию документации. Правила написания технологического задания к разрабатываемому программным продуктом. Техническое задание и требование к его содержанию. Эскизный и технический проекты. Рабочий проект. Внедрение.

Практическая работа №4 (2 ч)

1. Описание: совместная работа заказчика (говорит о пользе продукта, требованиях к работоспособности и внешнему виду) и EDISON (предлагает технические и алгоритмические решения).
2. Архитектура: утверждается язык программирования, база данных, серверы и фреймворки.
3. Техническое задание: составляется архитектором на основании описания и ответов заказчика на вопросы, согласовывается с менеджером проекта, затем передается клиенту, производятся правки.
4. Макеты (добавляются к техзаданию): интерфейсов, принципиальные схемы устройства, диаграммы структуры базы данных, схемы взаимодействия компонентов.
5. Контроль: архитектор устраняет замечания менеджера проектов.

6. Утверждение: заказчик проверяет и меняет ТЗ самостоятельно или сообщает список правок проект-менеджеру, замечания устраняются, ТЗ утверждается и прилагается к контракту.

Тема 1.6. Структура и формат статические и динамические данные (2 ч)

Понятие о ЕСПД. Виды программ. Виды программных документов. Виды эксплуатационных документов. Общие требования к программному документу. Обозначение программ и программных документов. Требования и правила для оформления структурных схем, алгоритмов. Понятие спецификации. Внешняя и внутренняя спецификации и их особенности. Требования к структуре внешней спецификации.

Раздел 2. Методология проектирование программных продуктов

Тема 2.1 Методы проектирования ПП (2 ч)

Методы проектирования программных продуктов и признаки их классификации. Неавтоматизированное и автоматизированное проектирование алгоритмов и программ.

Структурное проектирование программных продуктов и его методы. Принцип системного проектирования. Нисходящие проектирование. Модульное проектирование. Структурное программирование. Функционально-ориентированные методы и методы структурирования данных.

Информационное моделирование предметной области и его составляющие. Технологии информационного моделирования. Информационная и даталогические модели. Логический и физический уровень представления даталогической модели.

Сущность объектно-ориентированного подхода к проектированию программных продуктов. Объектно-ориентированный анализ предметной области и объектно-ориентированное проектирование. Объектно-ориентированная технология и ее преимущества.

Лабораторная работа №1 (2 ч)

Пример «Банкомат».

Диаграмма вариантов использования «Банкомат»

Оформление СЯС-карты

Клиент

Банк

Примеры С11С-карт

Примеры С11С-карт

Отчет по лабораторной работе должен состоять из:

1. Постановки задачи.
2. Описания действующих лиц и прецедентов системы.
3. Диаграммы прецедентов.
4. СЯС-карты.
5. Диаграммы взаимодействия.

Тема 2.2 Структура ПП (2 ч)

Внутренняя организация программного продукта. Цели структуризации программных продуктов. Типовая структура программного продукта. Головной, управляющий модуль, рабочие и сервисные модули. Структура пакетов прикладных программ.

Библиотеки стандартных программ и подпрограмм. Правила работы с библиотеками стандартных программ, встроенные функции. Возможность использования встроенных функций.

Лабораторная работа №2 (2 ч)

Системное ПО
Прикладное ПО
Инструментальные системы

Тема 2.3 Проектирования интерфейса пользователя (2 ч)

Интерфейс пользователя программного продукта. Классификация систем, поддерживающих диалоговые процессы. Системы с жестким сценарием, дескрипторные системы, тезаурусные системы, системы с языком деловой прозы. Характеристика сценария диалогового процесса. Требования, предъявляемые к стандартному графическому интерфейсу пользователя. Инструментарий создания интерфейса пользователя.

Лабораторная работа №3 (2 ч)

«Автоматизация работы организации по трудоустройству населения».
«Автоматизация работы транспортного агентства».
«Автоматизация обработки информации по работе туристической фирмы».
«Автоматизация учета реализации книжной продукции»
«Учет эмиграции населения».
«АРМ администратора ресторана».
«Автоматизация учебного процесса».

Раздел 3 Разработка программных продуктов

Тема 3.1 Стиль программирования (2 ч)

Понятия «стиль» и «стилистика» программирования. Правила хорошего стиля. Требования к стилю написания программы. Типы существующих стилей написания программы (классический, пользовательский, программиста и т.д.).

Лабораторная работа №4 (2 ч)

Краткие теоретические сведения

Стиль программирования связан с удобочитаемостью программы.

Правила хорошего стиля программирования – это результат соглашения между опытными программистами.

Правило стандартизации стиля заключается в следующем: если существует более одного способа сделать что-либо и выбор произвольный, остановитесь на одном способе, и всегда его придерживайтесь. Программное средство, представленное в хорошем стиле, имеет комментарии (пояснительные, вводные иногда оглавления), значимые идентификаторы, хорошо воспринимаемый текст ПС.

Пользовательский интерфейс также должен быть разработан в хорошем стиле, придерживаясь следующих рекомендаций:

- пользовательский интерфейс должен базироваться на терминах и понятиях, знакомых пользователю;
- пользовательский интерфейс должен быть единообразным;

- пользовательский интерфейс должен позволять пользователю исправлять собственные ошибки;
- пользовательский интерфейс должен позволять получение пользователем справочной информации: как по его запросу, так и генерируемой ПС.

Тема 3.2 Языки программирования (2 ч)

Практическая работа №5

Языки программирования и их классификация. Выбор и обоснование языка программирования. Языки программирования для решения экономических, научных, инженерных задач. Языки системного программирования. Комбинирование языков программирования в рамках одной задачи.

Современные языки программирования. общие понятия. Язык программирования Java. Общее сведение о языках программирования. Знакомства с интерфейсом программного языка Java. Создание первой программы «Привет мир!!!»

Тема 3.3 Модульное программирование (2 ч)

Лабораторная работа №5

Модульное программирование как метод разработки программ. Программный модуль и его основные характеристики. Типовая структура программного модуля. Порядок разработки программного модуля.

Задание:

- А) Оформить в виде основной программы и процедуры (функции) с параметрами программу, выполняющую обработку матрицы в соответствии с заданием лабораторной работы №2.
- В) Оформить в виде модуля программу, выполняющую обработку матрицы в соответствии с заданием лабораторной работы №2.

Тема 3.4 Структурное программирование (2 ч)

Теория и методы структурного программирования. Методы восходящей и нисходящей разработки структуры программы. Конструктивный и архитектурный подход к разработке программы.

Основные управляющие конструкции структурного программирования. Метод пошаговой детализации текста модуля. Структурное кодирования. Правила составления структурированных алгоритмов и их структурная композиция.

Основная концепция структурирования программ. Метод структурирования программ.

Практическая работа №6 (2 ч)

Структурное программирование.

Цель структурного программирования.

Цикл.

Бесконечный цикл.

Цикл с предусловием.

Цикл со счетчиком.

Задание 1 – Написать программу, выполненную в структурном стиле. Программа должна рассчитывать площадь фигур (программа должна корректно обрабатывать данные

согласно варианту в приложении А). На вход программа запрашивает строку, если в нее введено название фигуры, то программа запрашивает необходимые параметры фигуры, если введено значение отличное от названия фигуры, то программа повторно предлагает ввести название фигуры, если пользователь не справляется с этой задачей более 3 раз подряд, то программа сообщает о некорректности действий пользователя и завершается. В случае введения корректных данных программа должна выдать ответ, а также описание хода решения. Программа должна быть выполнена в виде блок-схемы и на ЯВУ.

Задание 2 – Написать программу вычисляющую площадь неправильного многоугольника. Многоугольник на плоскости задается целочисленными координатами своих N вершин в декартовой системе. Стороны многоугольника не соприкасаются (за исключением соседних - в вершинах) и не пересекаются. Программа в первой строке должна принимать число N – количество вершин многоугольника, в последующих N строках – координаты соответствующих вершин (вершины задаются в последовательности против часовой стрелки). На выход программа должна выдавать площадь фигуры. Программа должна быть выполнена в виде блок-схемы и на ЯВУ.

Тема 3.5 Объектно-ориентированное программирование

Практическая работа №7 (2 ч)

Основные понятия объектно-ориентированного проектирования. Объект, свойства объекта, метод обработки, событие, класс объектов.

Методика объектно-ориентированного проектирования и его основные принципы. Инкапсуляция, наследование, полиморфизм. Основные составляющие объектно-ориентированного анализа. Этапы объектно-ориентированного проектирования. Структура объектно-ориентированных программ.

1. Изучить теоретический материал.
2. Написать программу согласно индивидуальному варианту задания.
3. Ответить на контрольные вопросы.

Лабораторная работа №6 (2 ч)

ЗАДАНИЕ: Разработать объектно-ориентированную программу для вычисления длины любых отрезков прямых линий. Отрезки должны задаваться парами координат своих концов на плоскости через поток стандартного ввода. Результат вычисления длины каждого заданного отрезка должен отображаться через поток стандартного вывода. Программная реализация этих вычислений должна быть основана на разработке контейнерного класса отрезка прямой. Его компонентные данные должны включать подобъекты класса точки для хранения координат концов отрезка, а компонентный метод должен обеспечивать вычисление длины отрезка по ним. Конструкторы классов отрезка и точки должны использовать списки инициализации своих компонентных данных.

Тема 3.6 Эффективность и оптимизация программ (2 ч)

Понятие эффективности программ. Основные критерии эффективности программного продукта. Организация эффективной работы программы при экономичном использовании ресурсов ПЭВМ. Возможности увеличения быстродействия. Оптимизация программ на этапе отладки. Принципы и приемы оптимизации. Работа с оптимизирующими компиляторами.

Практическая работа №8 (2 ч)

Задание 1. Создание исходной таблицы.

- Задание 2. Сортировка данных.
Задание 3. Фильтрация. Задание
4. Итоги.

Тема 3.7 Обеспечение качества программного продукта (2 ч)

Принципы обеспечения показателей качества программного продукта. Функциональность и надежность как обязательные критерии качества программного продукта. Корректность программ, ее составляющие, программные эталоны и методы проверки корректности. Обеспечение легкости применения продукта. Обеспечение мобильности, модифицируемости и интеграции программных продуктов.

Лабораторная работа №7 (2 ч)

Цель работы: в лабораторной работе оцениваем качественные показатели ПП.

Раздел. 4 Отладка, тестирование и сопровождение программ

Тема 4.1 Ошибки программного обеспечения

Лабораторная работа №8,9 (4 ч)

Понятие об ошибке программного обеспечения. Источники ошибок программного обеспечения. Классификации ошибок программного обеспечения. Основные пути и методы борьбы с ошибками программного обеспечения. Обнаружение и локализация ошибок ввода и обработки данных.

1. Технические ошибки
2. Программные ошибки
3. Алгоритмические ошибки
4. Системные

Тема 4.2 Отладка программ

Лабораторная работа №10,11 (4 ч)

Понятие отладки программ. Составляющие процесса отладки. Принципы и виды отладок. Автономная и комплексная отладки программ. Методы отладки Средства отладки. Рекомендации по организации отладки Автономная отладка модуля. Использование средств отладки.

Отладка модулей программы и программы в целом.

Отладчик IntelliJ IDEA

1. *Запуск отладчика*

После того как вы настроите конфигурацию запуска вашего проекта, вы можете запускать его в режиме отладки, нажав *Shift + F9*.

В окне отладчика вы можете видеть стек вызовов функций и список потоков, с их состояниями, переменными и окнами просмотра состояния. Когда вы выбираете контекст вызова функции, вы можете просмотреть значения переменных соответствующих выбранному контексту.

2. *Полезные клавиатурные сокращения отладчика*

- o Установить/снять точку останова - *Ctrl + F8 (Cmd + F8 для Mac)*
- o Возобновить выполнение программы - *F9*
- o Перейти к следующей инструкции - *F8*

- o Перейти внутрь функции - *F7*
- o Приостановить выполнение - *Ctrl + F2 (Cmd + F2)*
- o Переключить между просмотром списка точек останова и подробной информацией о выбранной точке - *Shift + Ctrl + F8 (Shift + Ctrl + F8)*
- o Запустить отладку кода с точки на которой стоит курсор - *Shift + Ctrl + F9* (если это внутри метода *main()*)

3. Умный переход внутрь

Иногда вам надо при пошаговой отладке перейти внутрь определенного метода, но не первого который будет вызван. В таком случае вы можете нажать *Shift + F7 (Cmd + F7 для Mac)* чтобы выбрать из предложенного списка метод который вам нужен. Это может сэкономить вам массу времени.

Тема 4.3 Тестирование программ

Лабораторная работа №12,13 (4 ч)

Сущность и необходимость тестирования программного обеспечения. Различие между тестированием и отладкой программного обеспечения.

Основные принципы организации тестирования. Стадии тестирования. Виды тестовых проверок. Объекты тестирования и категории тестов. Виды тестирования.

Методы структурного тестирования программного обеспечения. Принцип «белого ящика». Пошаговое и монолитное тестирование модулей. Нисходящее и восходящее тестирование программного обеспечения.

Методы функционального тестирования. Принцип «черного ящика». Метод эквивалентного разбиения. Метод анализа граничных условий. Метод функциональных диаграмм. Комбинированные методы тестирования.

Средства тестирования. Ручное и автоматизированное тестирование. Применение методов и инструментальных средств тестирования.

- постановка задачи для теста;
- проектирование теста;
- написание тестов;
- тестирование тестов;
- выполнение тестов;
- изучение результатов тестирования.

Тема 4.4 Сопровождение программ

Лабораторная работа №14,15 (4 ч)

Сопровождение программных продуктов, внесение изменений, обеспечение надежности при эксплуатации. Необходимая документация и предпродажная подготовка программных средств.

Технические требования (спецификации). Руководства специалиста по сопровождению. Руководства пользователя. Руководства по вводу в действие и инсталляции.

Тема 4.5. Защита программ

Практическая работа №9,10 (4 ч)

Основные понятия о защите программных продуктов. Методы защиты программных продуктов от несанкционированного доступа и копирования. Системы разграничения доступа. Криптографические методы защиты программных продуктов, их особенности. Аппаратные средства защиты программного продукта.

Правовые методы защиты программных продуктов. Патентная защита лицензионные соглашения.

Алгоритм реализации:

1. Запомнить текущее время;
2. Выполнить контрольный участок кода;
3. Запомнить текущее время и разность текущего и предыдущего запомненного времени;
4. Выполнить контрольный участок кода повторно;
5. Сравнить разность текущего времени и предыдущего запомненного текущего времени с предыдущей запомненной разностью;
6. Если разности совпадают, продолжить выполнение, иначе – выйти из программы.

Тема 4.6 Пакеты прикладных программ

Практическая работа №11, 12, 13 (6 ч)

Проблемно–ориентированные ППП. Офисные ППП. Коммуникационные ППП

Раздел 5. Инструментальные средства разработки программ

5.1 Общая характеристика инструментальных средств разработки программ

Практическая работа №14 (2 ч)

Общая характеристика инструментальных средств разработки программ. Инструменты разработки программных продуктов. Инструментальные системы технологии программирования и их черты: комплексность, ориентированность на коллективную разработку, технологическая определенность, интегрированность. Основные компоненты инструментальных систем технологии программирования: репозиторий, инструментарий, интерфейсы.

CASE-средства, их назначение и применение. Классификации CASE-средств. Характеристика современных CASE-средств.

Текстовые редакторы. Интегрированные среды разработки. Компиляторы. Интерпретаторы. Линковщики. Парсеры и генераторы парсеров (см. Javacc). Ассемблеры. Отладчики. Профилировщики. Генераторы документации. Средства анализа покрытия кода. Средства непрерывной интеграции. Средства автоматизированного тестирования. Системы управления версиями и др.

Тема 5.2 Применение CASE-средств

Практическая работа №15 (2 ч)

Построение модулей программных систем с использованием структурного и объектно-ориентированного подхода. Диаграммы потоков данных и диаграммы «сущность связь».

Основные сведения о языке UML. Построение концептуальной модели предметной области. Диаграммы моделирования языка UML. Работа в среде CASE-средства.

Построение диаграмм потоков данных.

Раздел 6. Коллективная разработка программных средств (2 ч)

Тема 6.1 Организации работ при коллективной разработке программных продуктов

Категории специалистов, занятых разработкой и эксплуатацией программ. Принципы и методы коллективной разработки программных продуктов. Организация коллективной работы программистов. Схема взаимодействия специалистов, связанных с созданием и эксплуатацией программ. Типы организации бригад. Бригада главного программиста. Обязанности членов бригады. Распределение обязанностей в бригаде.

Тема 6.2 Экономические аспекты создания и использования программных средств

Стоимость программных средств. Факторы, влияющие на стоимость программных средств. Методики оценки трудоемкости разработки программного продукта. Особенности продаж программных продуктов. Обновление версий программных средств. Способы прогнозирования рынка программного обеспечения.

Раздел 7. Курсовое проектирование (18 ч)

Курсовое проектирование является завершающим этапом в изучении дисциплины «Технология разработки программных продуктов», в ходе которого осуществляется обучение применению полученных знаний и умений при решении комплексных задач, связанных со сферой профессиональной деятельности будущих специалистов.

Целью курсового проектирования является закрепление и углубление теоретических знаний, и приобретение практических навыков по разработке и проектированию ПО для заданной проблемы. Основными задачами курсового проекта являются:

- анализ возможных подходов и методов решения с обоснованием, выбранного подхода; выбор или разработка модели (математической, структурной, информационной), необходимой для достижения цели;
- выбор или эффективных алгоритмов с учетом их точности, устойчивости, сходимости; анализ полученных результатов работы программного обеспечения. систематизации и закрепления полученных теоретических знаний и практических умений по общепрофессиональным и специальным дисциплинам;
- углубления теоретических знаний в соответствии с заданной темой;
- формирования умения применять теоретические знания при решении поставленных профессиональных задач;
- формирования умения использовать справочную, нормативную и правовую документацию;
- развития творческой инициативы, самостоятельности, ответственности и организованности;
- подготовки к итоговой государственной аттестации.

1. Условия реализации учебной дисциплины.

3.1. Требования к минимальному материально-техническому обеспечению.

Реализация программы дисциплины требует наличие учебного кабинета «Информационная технология»

Методические указания:

1. Методические указания: по выполнению курсовой работы по дисциплине: «Технология разработки программных продуктов» Разработал: преподаватель информационных технологий:

Н.В. Данилова Челябинск 2017г.

2. методические указания по выполнению практической работы по дисциплине «Технология разработки программных продуктов» Новороссийских 2019г. Автор: Николаенко Т.П. – преподаватель НКРП

3. Системное программное обеспечение (СПО). (методические указания) Рубан В.И. Москва 2015г.

Наглядные пособия:

1. Плакат «Типичный процесс разработки ПО»
2. Плакат «процесс жизненного цикла ПО»
3. Презентация на тему: «Проектирование и разработка интерфейса ПО»
4. Видео по Питону: «Преимущества и недостатки программного языка Питон»

Технические средства обучения

1. Компьютер
2. Проектор
3. Доска интерактивная

3.2. Технические средства обучения:

Реализация учебной дисциплины требует наличия учебного кабинета оборудованного ТСО

Оборудование учебного кабинета:

- посадочные места по количеству студентов;
- рабочее место преподавателя;
- методические материалы по курсу дисциплины (курс лекций, методические рекомендации по подготовке к занятиям, дидактические единицы по дисциплине);
- компьютер;
- лицензионное программное обеспечение: Java, Notepad++, Sublime Text, Браузеры, Internet Explorer;
- интерактивная доска;
- проектор;
- комплекс мультимедиа – презентация по курсу дисциплины.

3. Контроль и оценка результатов освоения учебной дисциплины.

Критерии оценки выполнения студентами отчетных работ

Оценка "5" (отлично ставится в случае, если студент показывает правильное понимание сущности изучаемых ситуаций и закономерностей, методов и принципов; дает точное определение и истолкование основных понятий, принципов и методов;

указывает все свойства тех или иных объектов изучения: выполняет работу полностью, без ошибок и недочетов, с указанием всех необходимых свойств, законов, пояснений; схемы, графики, диаграммы выполнены точно; сделаны необходимые выводы.

Оценка "4" (хорошо) ставится, если работа студента от основным требованиям к работе на оценку "5" но в ней допущены одна ошибка или не более двух недочётов допущены ошибки в диаграммах: работа выполнена небрежно расчетных данных сделаны недостаточно полно.

Оценка "3" (удовлетворительно) ставится, если студент правильно понимает сущность изучаемых методов, понятий, теорем, законов, но в знаниях имеются пробелы, не мешающие выполнению основных требований, предусмотренных программой; если студент правильно выполнил не менее 2/3 всей работы или допустил не более одной грубой ошибки, не более трех негрубых ошибок, одной негрубой ошибки и трех недочетов, при наличии четырех-пяти недочетов.

Оценка "2" (неудовлетворительно) ставятся если студент выполнил менее 2/3 работы или допустил больше ошибок и недочетов, чем необходимо для "3"; не усвоил основные закономерности и понятия по курсу учебной дисциплины.

2. Основная литература

1. Иванова Г.С. Технология программирование: Учебник для вузов. М.: Изд-во МГТУ им. Н.Э. Баумана, 2020г.-320с.
2. Информатика: Учебник. – 3-е изд. перераб., /Под ред. Н.В. Макаровой. М.: Финансы и статистика, 2018г.
3. Орлов. В.В. Технологии разработки программных продуктов. СПб.: Питер, 2019-г.-437с.
4. Эрик Дж. Брауде. Технология разработки программного обеспечения. СПб.: Питер, 2020-г.-658с.
5. А.В. Рудаков Технология разработки программных продуктов. 2-е изд.-М.: изд. центр «Академия» 2019.г. 210с.

Дополнительная литература:

1. Стандарт IEEE по языку функционального моделирования - Синтаксис и семантика IDFF0. – Введ. 2016г. Нью-Йорк : IEEE, 2016г.
2. Стандарт IEEE по синтаксису и семантике языка концептуального моделирования IDEFIX(IDEF Object). - Введ. 2015г. Нью-Йорк : IEEE, 2015г.
3. Информационная технология – Процессы жизненного цикла программных средств. – Введ. 2013г. – Женева: ISO/IEC,2013г.
4. Системная и программная инженерия – Процессы жизненного цикла программных средств. – Введ. 2013г. – Нью-Йорк: ISO/IEC- IEEE,2013г.
5. Информационная технология – Оценка программного продукта – Ч.1: Общий обзор.
6. Программная инженерия – Качество продукта Ч.1: Модель качества. – Введ. 2014г. Женева: ISO/IEC,2014г.
7. Информационная технология. Классификация программных средств. – Введ. 2017г. М.: Издательство стандартов, 2016.

8. Информационная технология. Руководство по применению ГОСТ Р ИСО/МЭК 12207 (Процессы жизненного цикла программных средств). -М.: Изд-во стандартов, 2014г.
9. Информационные технологии. Процессы жизненного цикла программных средств. Минск: Госстандарт Респ. Беларусь, 2015.г
10. Информационные технологии. Оценки программной продукции. Характеристики качества и руководство по их – применению. – Минск: Госстандарт Респ. Беларусь, 2016.г
11. Информационные технологии поддержки жизненного цикла продукции. Методология функционального моделирование. Рекомендации по стандартизации. – Введ . 2016г.