

**УНИВЕРСИТЕТ «ТУРАН»
ДЕПАРТАМЕНТ МАГИСТРАТУРЫ И ДОКТОРАНТУРЫ**

Утверждаю
Проректор по внешним связям,
международной аккредитации и
поствузовскому образованию
учреждения «Университет «Туран»
д.э.н., профессор Тусупова Л.А.

Рассмотрено и одобрено
На заседании кафедры «Компьютерная и
программная инженерия»
Протокол № от «___»_____2021 г.

ПРЕРЕКВИЗИТЫ

ОП «Вычислительная техника и программное обеспечение»

по направлению подготовки

7М061 «Информационные и коммуникационные технологии»

Алматы, 2021

Содержание

1.	Пререквизит I – дисциплина «Системное программирование» - 3 кредита	
1.1	Программа курса	
1.2	Экзаменационные вопросы	
1.3	Список литературы	
2	Пререквизит II – дисциплина «Инструментальные средства разработки программ» - 2 кредита	
2.1	Программа курса	
2.2	Экзаменационные вопросы	
2.3	Список литературы	
3	Пререквизит III – дисциплина «Объектно - ориентированное программирование» - 3 кредита	
3.1	Программа курса	
3.2	Экзаменационные вопросы	
3.3	Список литературы	

1 Пререквизит I – дисциплина «Системное программирование» - 3 кредита

1.1 Программа курса

Цель преподавания дисциплины «Системное программирование» изучение основных принципов и методов программирования в современных ОС. Дается толкование основных понятий: объекты ядра, процесс, поток, приоритеты, атрибуты безопасности, кучи, мьютексы, семафоры, события. Основное внимание уделяется системным службам ядра, включая файловую систему, управление процессами и потоками, взаимодействие между процессами и синхронизацию. Приводится описание системных функций современных ОС. Рассматриваются основные свойства наиболее важных функций и показывается как применять их в реальных программных ситуациях. Даются базовые понятия основополагающих знаний об основных методах, средствах, принципах программирования в Win32.

1.2 Экзаменационные вопросы

1. Операционные системы и их интерфейсы.
2. Структура ОС Windows. Общее описание структуры системы
3. Управление потоками. Состояния потока.
4. Диспетчеризация и планирование потоков
5. Определение потока. API функции для работы с потоками
6. Управление процессами. Процессы и потоки в Windows.
7. Приоритеты потоков и процессов.
8. Синхронизация потоков в Windows.
9. Объекты синхронизации потоков.
10. Управление памятью. Использование виртуальной памяти.
11. Архитектура управления памятью в Win32.
12. Виртуальная память. Алгоритмы замещения страниц.
13. Динамически распределяемая память. Управление памятью кучи.
14. Использование файловой системы.
15. Управление файлами.
16. Управление каталогами.
17. Методы получения атрибутов файлов и каталогов.
18. Реестр. Разделы реестра. Управление реестром.
19. Отображаемые в память файлы.
20. Концепция динамически подключаемых библиотек. Создание DLL.
21. Динамическая загрузка и отключение DLL. Использование DLL.
22. Стандартные устройства и консольный ввод-вывод.
23. Асинхронный ввод-вывод и порты завершения.
24. Структурная обработка исключений. Фреймовая обработка исключений.
25. Структурная обработка исключений. Финальная обработка исключений.
26. Безопасность объектов Win32. Цели системы безопасности. Права и привилегии.
27. Безопасность объектов Win32. Списки контроля доступа
28. Межпроцессное взаимодействие. Анонимные каналы.
29. Межпроцессное взаимодействие. Именованные каналы.
30. Межпроцессное взаимодействие. Почтовые ячейки.

1.3 Список литературы

1. Джонсон М. Харт. Системное программирование в среде Win32. – М.: Издательский дом “Вильямс”, 2010.

Ф УТ 710–20–16. Пререквизиты специальности: программа курса. СМК УТ. Издание первое

2. Ал Вильямс. Системное программирование в Windows 2000. – СПб. Питер, 2011.
3. Пирогов В.Ю. Ассемблер для Windows. - СПб.:БХВ-Петербург, 2010.
4. Побегайло А. Системное программирование в Windows. – СПб.: БХВ-Петербург, 2006.
5. API. Энциклопедия программиста: Пер. с англ. – К.:Издательство “ДиаСофт”, 2009.

2 Пререквизит II – дисциплина «Инструментальные средства разработки программ» - 2 кредита

2.1 Программа курса

Дисциплина «Инструментальные средства разработки программ» в цикле технологической подготовки студентов, является синтезирующей, обеспечивающей необходимым арсеналом инструментальных средств для разработки программного обеспечения.

В результате изучения дисциплины студенты должны знать:

- основные направления в области проектирования, разработки программных продуктов и набора инструментальных средств, обеспечивающих их жизненный цикл;
- международные и отечественные стандарты, используемые при разработке программных продуктов;
- классические и современные подходы к проектированию интерфейса и моделированию информационной структуры приложения.

2.2 Экзаменационные вопросы

1. Этапы процесса разработки программ.
2. В чем отличие моделей деятельности предприятия AS-IS и TO-BE.
3. Какова основная идея методологии SADT.
4. Что такое спецификация программного обеспечения.
5. Какие два способа применяются при построении STD.
6. Перечислите основные методы задания спецификаций процессов.
7. Какие отношения между классами не переходят в связи между экземплярами?
8. Дайте определение ассоциации. Чем она отличается от наследования?
9. Зачем нужны классы-ассоциации? Постройте собственный пример.
10. Агрегирование является: свойством ассоциации, свойством роли, отдельным отношением между классами UML?
11. Чем агрегирование похоже на наследование и чем отличается?
12. Что такое UML -пакет?
13. Чем пакеты UML близки к проектам и solutions Microsoft Visual Studio?
14. Что такое связи между объектами?
15. Что такое кооперация? Приведите свой пример.
16. Перечислите основные абстракции моделирования структуры CPB.
17. Что такое интерфейс?
18. Что является предметной областью для моделей ПО?
19. Чем является UML: (i) предметной областью, (ii) моделью (iii) метамоделью (iv) метаметамоделью? Ответ обоснуйте.
20. Зачем для визуальных моделей выделять граф модели и диаграммы?
21. Что такое репозиторий CASE-пакета? Расскажите о способах его реализации.
22. Зачем для визуальных моделей выделять граф модели и диаграммы?
23. Каковы этапы разработки инфраструктуры семейства?

24. Что такое проектирование?
25. Расскажите о структуре типового DSM-пакета.
26. Что такое шаблон MVC? Физическая модель программной системы.
27. Зачем используются в GMF следующие модели: доменная, графическая, инструментальная?
28. Цели разработки диаграммы компонентов.
29. Элементы диаграммы компонентов. Компоненты.
30. Элементы диаграммы компонентов. Зависимости.

2.3 Список литературы

1. Мацяшек, Лешек, А. Анализ требований и проектирование систем. Разработка информационных систем с использованием UML: Перс англ.-М.: Издательский дом «Вильямс», 2002г.-432с.
2. Леффингуэлл, Дин, Уидриг, Дин. Принципы работы с требованиями к программному обеспечению. Унифицированный подход: Пер. с англ.-М.: Издательский дом «Вильямс»,2002г.-448с.
3. Элистер Кокберн. Современные методы описания функциональных требований к системам: Пер. с англ.- М.: Издательство «Лори»,2002г.-263с.
4. Амблер С.Гибкие технологии: экстремальное программирование и унифицированный процесс разработки. Библиотека программиста - СПб.: Питер, 2005г.412с.
5. Розенберг Д., Скотт К. Применение объектного моделирования с использованием UML и анализ прецедентов: Пер.с англ.-М.: ДМК Пресс, 2002г.-160с.
6. Кролл П., Крачтен Ф. Rational Unified Process –это легко. Руководство по RUP. Пер. с англ.-М.: КУДИЦ-ОБРАЗ, 2004г.-432с.
7. Орлов С.А. Технологии разработки программного обеспечения. Учебное пособие. 2-е изд.- СПб.: Питер,2003г.-480с.
8. Кубеков Б.С. Технологии разработки программного обеспечения. Учебник. – Алматы: Экономика, 2011. – 278с.

3 Пререквизит III – дисциплина «Объектно – ориентированное программирование» - 3 кредита

3.1 Программа курса

Углубленное изучение и освоение языков программирования на базе технологии объектно-ориентированного и обобщенного (с использованием стандартной библиотеки) программирования. В качестве базового используется язык программирования высокого уровня C++.

По завершении изучения дисциплины студент должен:

- освоить основные понятия, свойства и принципы объектно-ориентированного программирования;
- уметь проектировать спецификацию класса, выбирать целесообразную иерархию классов;
- освоить стандартную библиотеку языка C++;
- уметь использовать методику и средства технологии обобщенного программирования;
- уметь проектировать, специфицировать и отладить программу с оформлением документации и использованием диаграммной техники языка UML.

3.2 Экзаменационные вопросы

1. Понятие объекта. Характерные свойства объекта. Как это используется в ООП.
2. Структура объекта. Объект с точки зрения реализации в языке программирования. Приведите пример для любого языка программирования.
3. Классы объектов. Какие задачи они решают. Метаклассы.
4. Структурные отношения между объектами.
5. Ассоциативный класс (класс-ассоциация), какие задачи он решает.
6. События. Виды событий. Распространение информации о событии.
7. Косвенная рекурсия при распространении событий. Существующие способы избегать косвенной рекурсии.
8. Время ОО-системы. Направления специализации ОО-систем по способу организации их времени.
9. Состояние системы. Степень сложности системы. Принцип транзакций.
10. Объект как совокупность служб/сервисов. Интерфейс/протокол объекта.
11. Инкапсуляция. Области видимости в ООП.
12. Отношение обобщения и свойство наследования. Подходы разработки "от конкретного к абстрактному" и "от абстрактного к конкретному".
13. Множественное наследование.
14. Полиморфизм. Свойство полиморфизма. Реализация механизма полиморфизма в языках программирования (например, в C++).
15. Иллюстрация полиморфизмом разницы между отправкой сообщения объекту и вызовом его метода.
16. Абстрактный метод. Ограничения, накладываемые на класс.
17. Класс-интерфейс (интерфейсный класс), какие задачи решает. Интерфейсный класса в иерархии.
18. Класс, объекты.
19. Атрибут (свойство) и поведение объекта.
20. Отношение наследования в следующих классах: классы Двигатель и Дизель или классы Двигатель и Автомобиль.
21. Конструктор и деструктор.
22. Конструкторы, деструкторы и наследование
23. Категория защищенности protected.
24. Различие между обычным и виртуальным базовыми классами.
25. Пользовательская функция вывода. Пользовательская функция ввода.
26. Особенности реализации ООП в C++.
27. Переменная экземпляра.
28. Список аргументов и список параметров.
29. Конструктор и его определение.
30. Конструктор копирования и его использование.

3.3 Список литературы

1. Т. А. Павловская «С/С++. Программирование на языке высокого уровня» СПб.: Питер, 2018.-496 с:ил.
2. Павловская Т. А., Щупак Ю. А. С++. Объектно-ориентированное программирование: Практикум. — СПб.: Питер, 2006. — 265 с: ил.
3. Иванова Г. С. Объектно-ориентированное программирование : учебник / Г. С. Иванова, Т. Н. Ничушкина ; под общ. ред. Г. С. Ивановой. — М. : Изд-во МГТУ им. Н. Э. Баумана, 2014. — 455, [1] с. : ил.

4. Шлее М. Профессиональное программирование на С++. СПб.: БХВ-Петербург, 2010.
5. Шилдт Г. Полный справочник по С++. 4-е изд. М.: «Вильямс», 2009

Разработано

Зав. кафедрой «Компьютерная и
программная инженерия» _____

PhD Жасандықызы М

Согласовано

Декан факультета «Цифровые
технологии и искусство» _____

к.т.н., профессор Куандыкова Д.Р.