

Лабораторная работа № 4

Тема: разработка моделей информационных систем по методологии моделирования процессов (IDEF3)

Цель: изучить методологию моделирования процессов (IDEF3), освоить CASE-средство AllFusion Process Modeller для построения диаграмм IDEF3, научиться строить модели информационных систем, используя данную методологию

Краткая теория

Методология IDEF3 – способ описания процессов, основной целью которого является обеспечение структурированного метода, используя который эксперт в предметной области может описать положение вещей как упорядоченную последовательность событий с одновременным описанием объектов, имеющих непосредственное отношение к процессу. Данная методология хорошо приспособлена для сбора данных, требующихся для проведения структурного анализа системы. В отличие от большинства технологий моделирования бизнес-процессов, IDEF3 не имеет жестких синтаксических или семантических ограничений, делающих неудобным описание неполных или нецелостных систем. Технология IDEF3 также может быть использована как метод проектирования бизнес-процессов. IDEF3-моделирование органично дополняет традиционное моделирование с использованием методологии IDEF0.

Основой модели IDEF3 служит так называемый сценарий бизнес-процесса, который выделяет последовательность действий или подпроцессов анализируемой системы. Так как сценарий определяет назначение и границы модели, довольно важным является подбор подходящего наименования для обозначения действий. Для подбора необходимого имени применяются стандартные рекомендации по предпочтительному использованию глаголов и отглагольных существительных, аналогично методологии IDEF0. Также как и в методологии IDEF0, для модели IDEF3 должна быть явным образом документирована точка зрения. А для системного аналитика также важно понимание цели моделирования – набора вопросов, ответами на которые будет служить модель, границ моделирования (какие части системы войдут в модель, а какие не будут в ней отображены) и целевой аудитории (для кого разрабатывается модель).

Главной организационной единицей модели IDEF3 является диаграмма. Основным элементом, изображаемым на диаграммах IDEF3, является «единица работы» (Unit of Work - UOW). Диаграммы IDEF3 отображают действие в виде прямоугольника. Как уже отмечалось, действия именуются с использованием глаголов или отглагольных существительных, каждому из действий присваивается уникальный идентификационный номер. Этот номер не используется вновь даже в том случае, если в процессе построения модели действие удаляется. В диаграммах IDEF3 номер действия обычно предваряется номером его родителя (рисунок 1).



Рисунок 1 – Изображение и нумерация действия на диаграмме IDEF3

Связи выделяют существенные взаимоотношения между действиями. Все связи в IDEF3 являются однонаправленными, и, хотя стрелка может начинаться или заканчиваться на любой стороне блока, обозначающего действие, диаграммы IDEF3 обычно организовываются слева

направо таким образом, что стрелки начинаются на правой и заканчиваются на левой стороне блоков. В таблице 4.1 приведены три возможных типа связей.

Таблица 4.1 – Типы связей в модели IDEF3

Название	Изображение	Назначение
Временное предшествование	→	Исходное действие должно завершиться прежде, чем конечное действие может начаться
Объектный поток	⇒	Выход исходного действия является входом конечного действия. Из того, в частности, следует, что исходное действие должно завершиться прежде, чем конечное действие сможет начаться
Нечеткое отношение	- - - - ->	Вид взаимодействия между исходным и конечным действиями задается аналитиком отдельно для каждого случая использования такого отношения

Связь типа "Временное предшествование". Как видно из названия, связи этого типа отражают, что исходное действие должно полностью завершиться, прежде чем начнется выполнение конечного действия. Связь должна быть поименована таким образом, чтобы человеку, просматривающему модель, была понятна причина ее появления. Во многих случаях завершение одного действия инициирует начало выполнения другого, как показано на рисунок 2. В этом примере автор должен принять рекомендации рецензентов, прежде чем начать вносить соответствующие изменения в работу.

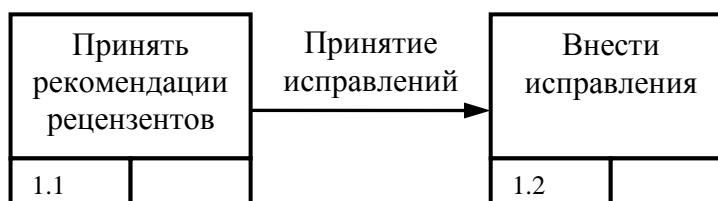


Рисунок 2 – Пример связи «Временное предшествование»

Связь типа "Объектный поток". Одной из наиболее часто встречающихся причин использования связи типа "объектный поток" состоит в том, что некоторый объект, являющийся результатом выполнения исходного действия, необходим для выполнения конечного действия. Такая связь отличается от связи временного предшествования двойным концом обозначающей ее стрелки. Наименования потоковых связей должны четко идентифицировать объект, который передается с их помощью. Временная семантика объектных связей аналогична связям предшествования. Это означает, что порождающее конечное действие начнет выполняться, как показано на рисунке 3. В приведенном примере счет на оплату услуг является результатом выполнения действия 1.1. Счет необходим для проведения оплаты услуг.



Рисунок 3 – Пример связи «Объектный поток»

Связь типа "Нечеткое отношение". Связи этого типа используются для выделения отношений между действиями, которые невозможно описать с использованием

предшественных или объектных связей. Значение каждой такой связи должно быть определено, поскольку связи типа "Нечеткое отношение" сами по себе не предполагают никаких ограничений. Одно из применений нечетких отношений — отображение взаимоотношений между параллельно выполняющимися действиями. Рисунок 4 иллюстрирует фрагмент процесса запуска бензопилы с водяным охлаждением и нечеткое отношение между действиями "Запустить двигатель" и "Запустить водяной насос". Название стрелки может быть использовано для описания природы отношения, более подробное объяснение может быть приведено в виде отдельной ссылки.

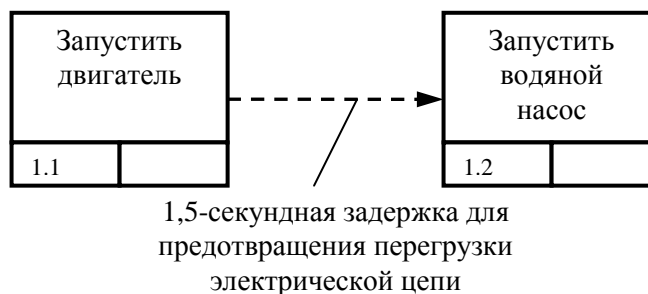


Рисунок 4 – Пример связи «Нечеткое отношение»

Наиболее часто нечеткие отношения используются, если последующее действие может начать свое выполнение, после начала выполнения предшествующего действия, но до его завершения. При этом последующее действие может завершиться ранее предшествующего действия.

Завершение одного действия может инициировать начало выполнения сразу нескольких других действий, или, наоборот, определенное действие может требовать завершения нескольких других действий для начала своего выполнения. Соединения разбивают или соединяют внутренние потоки и используются для описания ветвления процесса.

–Разворачивающие соединения используются для разбиения потока. Завершение одного действия вызывает начало выполнения нескольких других.

–Сворачивающие соединения объединяют потоки. Завершение одного или нескольких действий вызывает начало выполнения только одного другого действия.

В таблице 2 приведены три типа соединений.

"И"-соединения. Соединения этого типа инициируют выполнение всех своих конечных действий. Все действия, присоединенные к сворачивающему "И"-соединению, должны завершиться, прежде чем может начать выполняться следующее действие. На рисунке 5 после обнаружения пожара инициируются включение пожарной сигнализации, вызов пожарной охраны и начинается тушение пожара. Запись в журнал производится только тогда, когда все три перечисленных действия завершены.

Соединение "Исключающее ИЛИ". Вне зависимости от количества действий, прицепленных к сворачивающему или разворачивающему соединению "Исключающее ИЛИ", инициировано будет только одно из них, и поэтому только одно из них будет завершено перед тем, как любое действие, следующее за сворачивающим соединением "Исключающее ИЛИ", сможет начаться. Если правила активации соединения известны, они обязательно должны быть документированы либо в его описании, либо пометкой стрелок, исходящих из разворачивающего соединения, как показано на рисунке 6. На рисунке 6 соединение "Исключающее ИЛИ" используется для отображения того факта, что студент не может одновременно быть направлен на лекции по двум разным курсам.

Соединение "ИЛИ". Соединения этого типа предназначены для описания ситуаций, которые не могут быть описаны двумя предыдущими типами соединений. Аналогично связи нечеткого отношения соединение "ИЛИ" в основном определяется и описывается непосредственно системным аналитиком. На рисунке 7 соединение J2 может активировать проверку данных чека и (или) проверку суммы наличных. Проверка чека инициируется, если

покупатель желает расплатиться чеком, проверка суммы наличных – при оплате наличными. И то, и другое действие инициируется при частичной оплате чеком и частичной – наличными.

Таблица 2 – Типы соединений в модели IDEF3

Название	Обозначение	Вид	Правила инициации
Соединение «И»		Разворачивающее	Каждое конечное действие обязательно инициируется
		Сворачивающее	Каждое исходное действие обязательно должно завершиться
Соединение «Исключающее ИЛИ»		Разворачивающее	Одно и только одно конечное действие инициируется
		Сворачивающее	Одно и только одно исходное действие должно завершиться
Соединение «ИЛИ»		Разворачивающее	Одно (или более) конечное действие инициируется
		Сворачивающее	Одно (или более) исходное действие должно завершиться

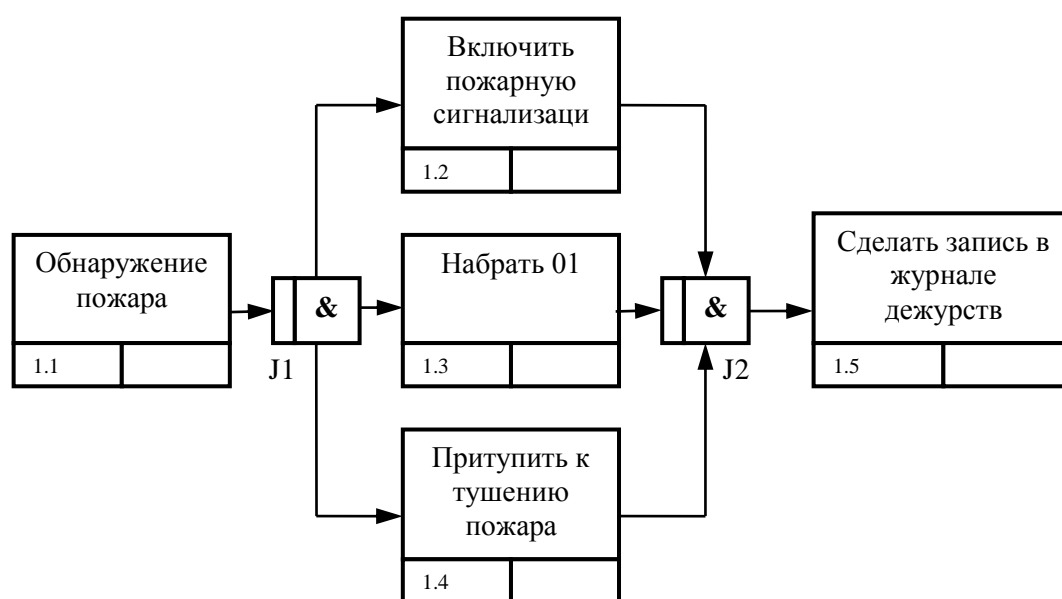


Рисунок 5 – Пример «И»-соединения

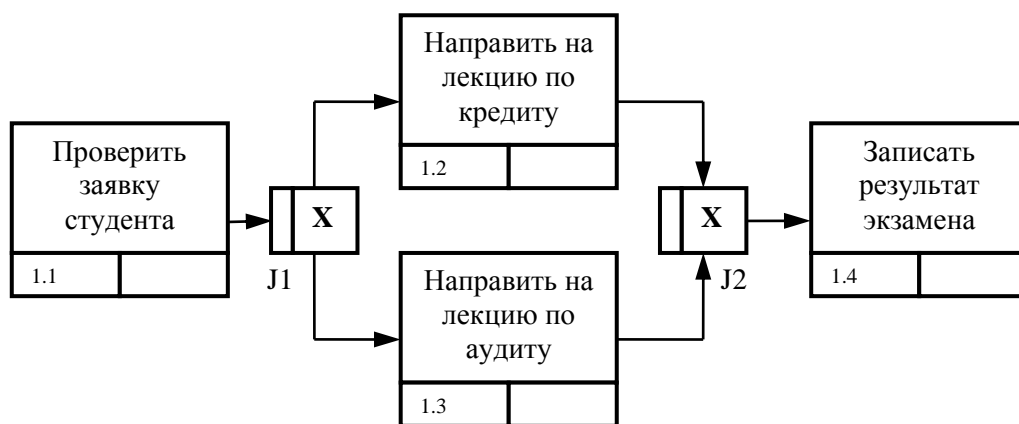


Рисунок 6 – Пример «Исключающее ИЛИ»-соединения

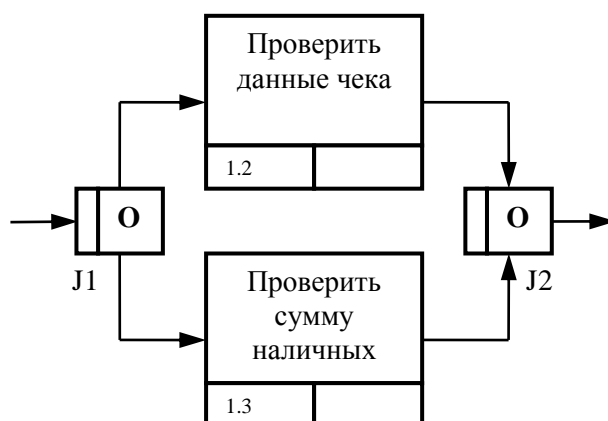


Рисунок 7 – Пример «ИЛИ»-соединения

В приведенных примерах все действия осуществлялись асинхронно, т.е. они не должны были начинать выполняться одновременно. Однако есть случаи, когда время начала или окончания параллельно выполняемых действий должно быть одинаковым, т.е. действия должны выполняться синхронно. Для моделирования такого поведения системы используются синхронные соединения. В таблице 3 приведены виды синхронных соединений.

Таблица 3 – Типы синхронных соединений в модели IDEF3

Название	Обозначение	Вид	Правила инициации
Соединение «И»		Разворачивающее	Все действия начнутся одновременно
		Сворачивающее	Все действия закончатся одновременно
Соединение «ИЛИ»		Разворачивающее	Несколько действий, возможно, начнутся одновременно
		Сворачивающее	Несколько действий, возможно, завершатся одновременно

Синхронное соединение обозначается двумя вертикальными линиями внутри обозначающего его прямоугольника в отличие от одной вертикальной линии в асинхронном соединении. На рисунке 8 приведен пример использования синхронного разворачивающего

«И»-соединения для ситуации начала состязаний, когда необходимо обеспечить одновременное начало действий: выстрел из стартового пистолета, запуск секундомера и начало забега.

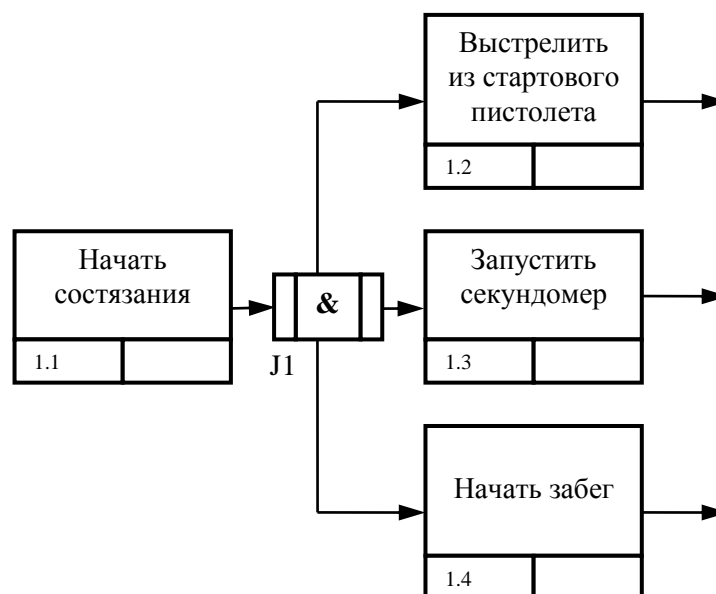


Рисунок 8 – Пример синхронного «И»-соединения

Все соединения на диаграммах должны быть парными, из чего следует, что любое разворачивающее соединение имеет парное себе сворачивающее. Однако типы соединений вовсе не обязательно должны совпадать. При этом синхронное разворачивающее соединение не обязательно должно иметь парное себе синхронное сворачивающее соединение и наоборот: асинхронное разворачивающее соединение необязательно имеет парное асинхронное соединение. На рис. 9 разворачивающее «И»-соединение имеет парное сворачивающее «ИЛИ»-соединение. Интерпретация соединения J1 аналогична случаю, показанному на рисунке 5. Соединение J2 интерпретируется следующим образом: после включения пожарной сигнализации и (или) вызова пожарных и (или) начала тушения производится запись в журнал.

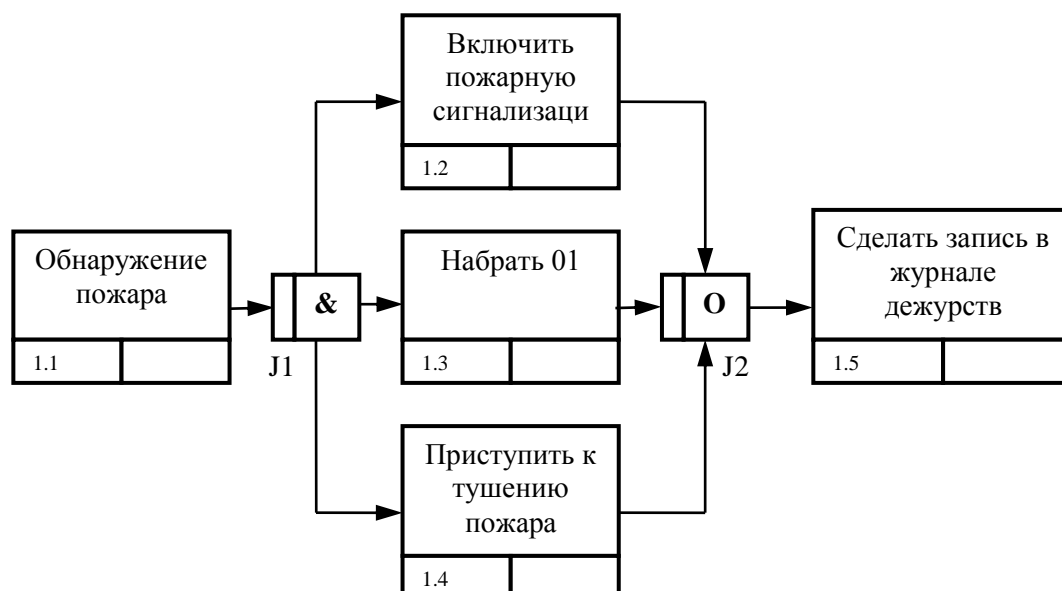


Рисунок 9 – Пример комбинации двух типов соединений

Соединения могут комбинироваться для создания более сложных правил ветвления (рисунок 10). Комбинации соединения следует использовать с осторожностью, поскольку перегруженные ветвлением диаграммы могут оказаться сложными для восприятия.

Помимо приведенных элементов на диаграммах IDEF3 могут изображаться указатели. Указатели — это специальные символы, которые ссылаются на другие разделы описания

процесса. Они выносятся на диаграмму для привлечения внимания читателя к каким-либо важным аспектам модели. Типы указателей и их применение приведены в таблице 4.4.

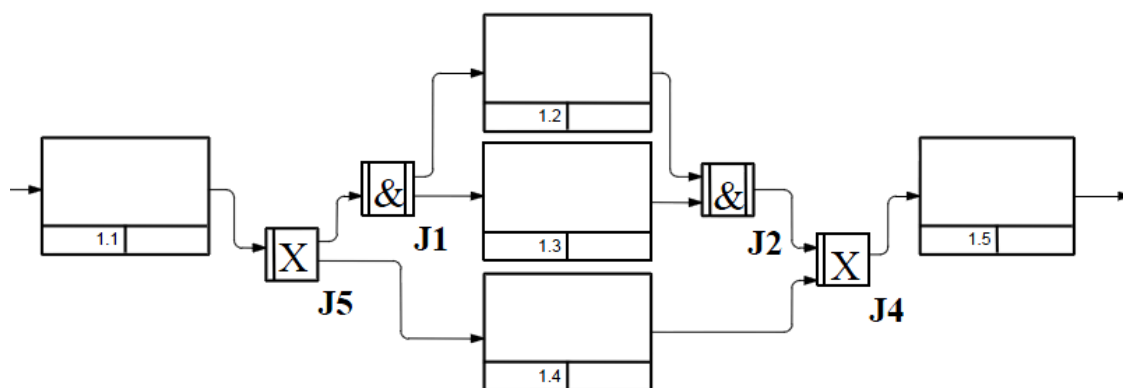


Рисунок 10 – Пример комбинации соединений

Таблица 4 – Типы указателей модели IDEF3

Тип указателя	Назначение
Объект (OBJECT)	Для описания того, что в действии принимает участие какой-либо заслуживающий отдельного внимания объект.
Ссылка (GOTO)	Для реализации цикличности выполнения действий. Указатель ССЫЛКА может относиться и к соединению.
Единица действия (Unit of behavior - UOB)	Для помещения на диаграмму дополнительного экземпляра уже существующего действия без заикливания.
Заметка (NOTE)	Для документирования любой важной информации общего характера, относящейся к изображенному на диаграммах.
Уточнение (Elaboration – ELAB)	Для уточнения или более подробного описания изображенного на диаграмме. Указатели УТОЧНЕНИЕ обычно используются для описания логики ветвления у соединений.

Указатель изображается на диаграмме в виде прямоугольника, похожего на изображение действия. Имя указателя обычно включает его тип (например, OBJECT, UOB и т.п.) и идентификатор. На рисунке 11 изображен указатель типа объект со связью с единицей действия.

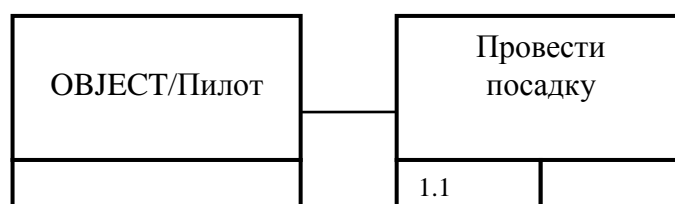


Рисунок 11 – Изображение указателя

Действия в IDEF3 могут быть декомпозированы, или разложены на составляющие, для более детального анализа. Декомпозировать действие можно несколько раз. Это позволяет документировать альтернативные потоки процесса в одной модели. Для корректной идентификации действий в модели с множественными декомпозициями схема нумерации действий расширяется и наряду с номерами действия и его родителя включает в себя порядковый номер декомпозиции. Например, в номере действия 1.2.5: 1 — номер родительского действия, 2 — номер декомпозиции, 5 — номер действия.

Ход работы

1. Получить вариант индивидуального задания.
2. Определить для проектируемого процесса единицы работы (OUW).
3. Определить виды взаимосвязей между процессами.
4. Построить диаграмму IDEF3, моделирующую процесс.

Варианты заданий:

1.	Смоделировать процесс изучения студентами дисциплины. Изучение дисциплины в течение семестра предполагает прослушивание курса лекций и выполнение курса из 4 лабораторных работ (работы могут выполняться в произвольном порядке). Дополнительно студент должен выполнить контрольную работу. Учесть, что время выдачи контрольной работы (начала ее выполнения) не совпадает с началом курсов лабораторных работ и лекций. Итоговой формой контроля изучения дисциплины является зачет и экзамен. Зачет выставляется после защиты всех лабораторных работ и сдачи контрольной работы. После получения зачета студент допускается к экзамену и сдает его. Если экзамен сдан неудовлетворительно, то студент может сдать его повторно. Если и повторная сдача проходит неуспешно, то студент сдает данный экзамен комиссии. После чего принимается решение о повторном прохождении курса.
2.	Необходимо смоделировать процесс сборки системного блока компьютера. Процесс сборки начинается с проверки наличия всех необходимых комплектующих. Сначала на материнскую плату монтируются основные компоненты: процессор и модули памяти. Они могут монтироваться в произвольном порядке независимо друг от друга. Затем собранная материнская плата монтируется в корпус. После чего на нее подключается видеокарта и разъемы питания (также независимо друг от друга в любом порядке). Далее осуществляется подключение внутренних устройств: жесткий диск и DVD-привод (они монтируют и подключаются в любом порядке независимо друг от друга).
3.	Смоделировать процесс изучения студентами дисциплины. Изучение дисциплины в течение семестра предполагает прослушивание курса лекций и выполнение курса из 4 лабораторных работ (работы могут выполняться только последовательно). Дополнительно студент должен выполнить курсовую работу. Учесть, что время выдачи курсовой работы (начала ее выполнения) не совпадает с началом курсов лабораторных работ и лекций. Итоговой формой контроля изучения дисциплины является экзамен. Студент допускается к сдаче экзамена, только после выполнения всех лабораторных работ и защиты курсовой работы. Если экзамен сдан неудовлетворительно, то студент может сдать его повторно. Если и повторная сдача проходит неуспешно, то студент сдает данный экзамен комиссии. После чего принимается решение о повторном прохождении курса.
4.	Необходимо смоделировать процесс установки программного обеспечения на только что собранный компьютер. Сначала выполняется подбор необходимого ПО. Затем устанавливается операционная система. После установки ОС осуществляется установка драйверов. После того, как ОС и драйвера установлены, осуществляется установка прикладного ПО. На компьютер необходимо установить: пакет MS Office, архиватор, аудио- и видео-проигрыватели, среду разработки программ (одну на выбор: MS Visual Studio, RAD Studio, Eclipse, NET Beans). Каждый вид прикладного ПО может устанавливаться независимо от других в произвольном порядке.
5.	Смоделировать процесс сдачи контрольных работ студентами заочной формы обучения. Процесс начинается с выдачи задания. Далее студент выполняет его. После того как контрольная работа выполнена студент сдает ее на кафедру, предварительно он может (необязательно) зарегистрировать ее в деканате. После сдачи работы она проверяется преподавателем. Если работа выполнена правильно, то она остается на кафедре с пометкой «К защите». Если работа содержит ошибки, то она возвращается студенту на доработку. Студент ее дорабатывает, и процесс сдачи на проверку повторяется.

6.	Смоделировать процесс разработки программного обеспечения. Разработка ПО начинается с системного анализа, после которого выполняется определение функциональных требований. Результатом последнего процесса становится техническое задание, которое передается на этап предварительного проектирования. На этом этапе выделяются модули, из которых состоит приложение (предположим что их 5). Разработка каждого модуля ведется независимо, но разработка всех модулей начинается в один тот же момент времени, а завершение разработки – в разные моменты времени. Параллельно с процессами предварительного проектирования и разработки модулей, выполняется процесс разработки тестов. После завершения разработки всех модулей и тестов последовательно выполняются тестирование интеграции и тестирование правильности. В завершении выполняется процесс внедрения разработанного ПО.
7.	Смоделировать процесс прохождения студентом преддипломной практики. Во время преддипломной практики студент сначала знакомится с местом прохождения практики. Далее он осуществляет выбор темы дипломного проектирования, после которого осуществляется сбор необходимых сведений для выполнения дипломного проектирования и разработка технического задания (могут осуществляться параллельно, но зависят друг от друга). Параллельно с работой над темой дипломного проекта выполняется сопутствующая работа на месте прохождения практики. В конце практики осуществляется составление отчета по практике и разработка технического задания (они могут выполняться параллельно, то разработка ТЗ зависит от составления отчета). В завершении всего выполняется защита преддипломной практики.
8.	Смоделировать процесс прохождения студентом теста по определенной дисциплине. В начале студент регистрируется в системе и ему предоставляется список дисциплин. Студент выбирает одну из дисциплин и заявка о выполнении теста отправляется преподавателю. Преподаватель может либо подтвердить заявку, либо отклонить ее. Если заявка подтверждена, то студенту сразу высылаются пять тестовых заданий. Выполнять тестовые задания студент может в произвольном порядке независимо друг от друга, в параллельном или последовательном режиме. После этого он отправляет результаты выполнения тестов, при этом он может выполнить не все задания, а только часть из них. Результаты проверяет преподаватель, и отправляет студенту результаты проверки и оценку за тест.
9.	Смоделировать процесс прохождения студентом дипломного проектирования. Во время дипломного проектирования выполняется разработка ПО, которая проходит за три этапа: проектирование, реализация и тестирование. После завершения каждого этапа начинается написание соответствующего раздела пояснительной записки. Параллельно с разработкой ПО и написанием основных разделов пояснительной записки студент осуществляет разработку разделов охрана труда, защита населения, экономика. Выполнение этих разделов осуществляется независимо друг от друга в произвольном порядке. После завершения разработки ПО и написания всех разделов пояснительной записки осуществляется предварительная защита дипломного проекта, по результатам которой студенту назначается рецензент. Дипломная работа проходит рецензирование и после этого допускается к защите. В конце производится защита дипломного проекта.
10.	Смоделировать процесс выполнения студентом комплекса лабораторных работ. Студент в процессе изучения некоторой дисциплины должен выполнить 8 лабораторных работ. Все лабораторные работы выполняются последовательно: задание на следующую работу студент может получить только после выполнения и защиты предыдущей. Выполнение каждой лабораторной работы состоит из двух этапов: выполнение задания и защита отчета. Результатом выполнения задания является отчет, который передается на этап защиты лабораторной работы. Если студент не защищает лабораторную работу, то ему выдается дополнительное задание по этой работе. Он его выполняет и снова защищает. Так продолжается до тех пор, пока студент не выполнит все лабораторные работы.

11.	Смоделировать процесс изучения студентом ряда дисциплин в течение семестра. Всего в семестре изучается 9 дисциплин. По пяти из них сдаются зачеты. После сдачи всех пяти зачетов студент допускается к сессии. На сессии студент должен сдать четыре экзамена по оставшимся дисциплинам. Если студент не сдает три экзамена, то он отчисляется. Учесть, что студент может быть отчислен до конца сессии: как только он не сдаст третий экзамен.
-----	---

Контрольные вопросы

1. Для каких целей предназначена методология IDEF3?
2. Что является основным элементом диаграммы IDEF3 и как он обозначается?
3. Какие виды связей могут использоваться между единицами действия?
4. В каком случае используется связь «Временное предшествование»?
5. В каком случае используется связь «Объектный поток»?
6. В каком случае используется связь «Нечеткое отношение»?
7. Для чего на диаграммах IDEF3 используются соединения?
8. Какие виды соединений могут присутствовать на диаграммах IDEF3?
9. Опишите нотацию и семантику асинхронного соединения «И».
10. Опишите нотацию и семантику асинхронного соединения «ИЛИ».
11. Опишите нотацию и семантику асинхронного соединения «Исключающего ИЛИ».
12. Опишите нотацию и семантику синхронного соединения «И».
13. Опишите нотацию и семантику синхронного соединения «ИЛИ».
14. Что такое указатель, и для каких целей он используется на диаграммах IDEF3?
15. Какие виды указателей бывают на диаграммах IDEF3?
16. Как осуществляется нумерация единиц работы на диаграммах IDEF3?