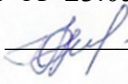


**МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ  
РОССИЙСКОЙ ФЕДЕРАЦИИ**  
**Федеральное государственное бюджетное образовательное учреждение  
высшего образования**  
**«Пермский государственный национальный исследовательский университет»**

*Колледж профессионального образования*

**ИНСТРУМЕНТАЛЬНЫЕ СРЕДСТВА РАЗРАБОТКИ  
ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ**

Методические рекомендации  
для самостоятельных работ по изучению дисциплины  
для студентов Колледжа профессионального образования  
специальности  
09.02.03 Программирование в компьютерных системах

Утверждено на заседании ПЦК  
Информационных технологий  
Протокол № 9 от 23.05.2018  
председатель  Н.А. Серебрякова

Пермь 2018

Составитель:

*Бочкарев Алексей Михайлович*, преподаватель первой квалификационной категории, преподаватель ПГНИУ

Инструментальные средства разработки программного обеспечения: методические указания по самостоятельной работе для студентов Колледжа профессионального образования по специальностям 09.02.03 Программирование в компьютерных системах и / сост. А.М. Бочкарев; Колледж проф. образ. ПГНИУ. – Пермь, 2020. – 23 с.

Методические указания «Инструментальные средства разработки программного обеспечения» разработаны на основе требований Федерального государственного образовательного стандарта среднего профессионального образования по специальностям 09.02.03 Программирование в компьютерных системах и для оказания помощи студентам специальностей 09.02.03 Программирование в компьютерных системах и по дисциплине «Инструментальные средства разработки программного обеспечения». Содержат самостоятельные задания по разделам дисциплины.

Предназначены для студентов Колледжа профессионального образования ПГНИУ специальностей 09.02.03 Программирование в компьютерных системах и (СПО) всех форм обучения.

## СОДЕРЖАНИЕ

<b>Компьютерная поддержка разработки и сопровождения программ- ных средств</b>	<b>4</b>
<b>Разработка приложений с помощью IDE</b>	<b>4</b>
<b>Основные классы инструментальных сред разработки и сопровождения программных средств</b>	<b>4</b>

# САМОСТОЯТЕЛЬНАЯ РАБОТА № 1

## Создание диаграммы декомпозиции.


### Цель работы:

Создание диаграммы декомпозиции для данного примера.

### Исходные данные (задание):

#### Создание диаграммы декомпозиции.

Перед выполнением этой работы внимательно прочитайте раздел «Работа» и раздел «Стрелки»

1. Выберите кнопку перехода  на нижний уровень в палитре инструментов, в диалоговом окне Activity Box Count установите число работ 3 на диаграмме нижнего уровня нажмите кнопку ОК (рис. 1.)

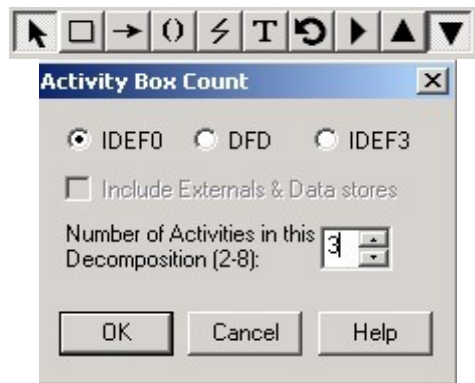


Рис.1.

Автоматически будет создана диаграмма декомпозиции. Правой кнопкой мыши щелкните по первой работе, выберите Name и внесите имя работы, например, «Продажи и маркетинг». Повторите операцию для всех трех работ. Затем внесите определение, статус и источник для каждой работы согласно табл. 2.

Табли-

ца 2. Описание работ для диаграммы декомпозиции

Функциональный блок (Name)	Описание (Definition)	Статус (Status)	Источник (Source)
Продажи, маркетинг	Телемаркетинг, презентации, выставки	WORKING	Материалы курса по BPwin
Сборка, тестирование компьютеров	Сборка и тестирование настольных и портативных компьютеров	WORKING	Материалы курса по BPwin

Отгрузка, получение	Отгрузка заказов клиентам и получение компонентов от поставщиков	WORKING	Материалы курса по BPwin

2. Для изменения свойств работ после их внесения в диаграмму можно воспользоваться словарем объектов модели. Вызов словаря **Model/Diagram Object Editor** (рис. 2)

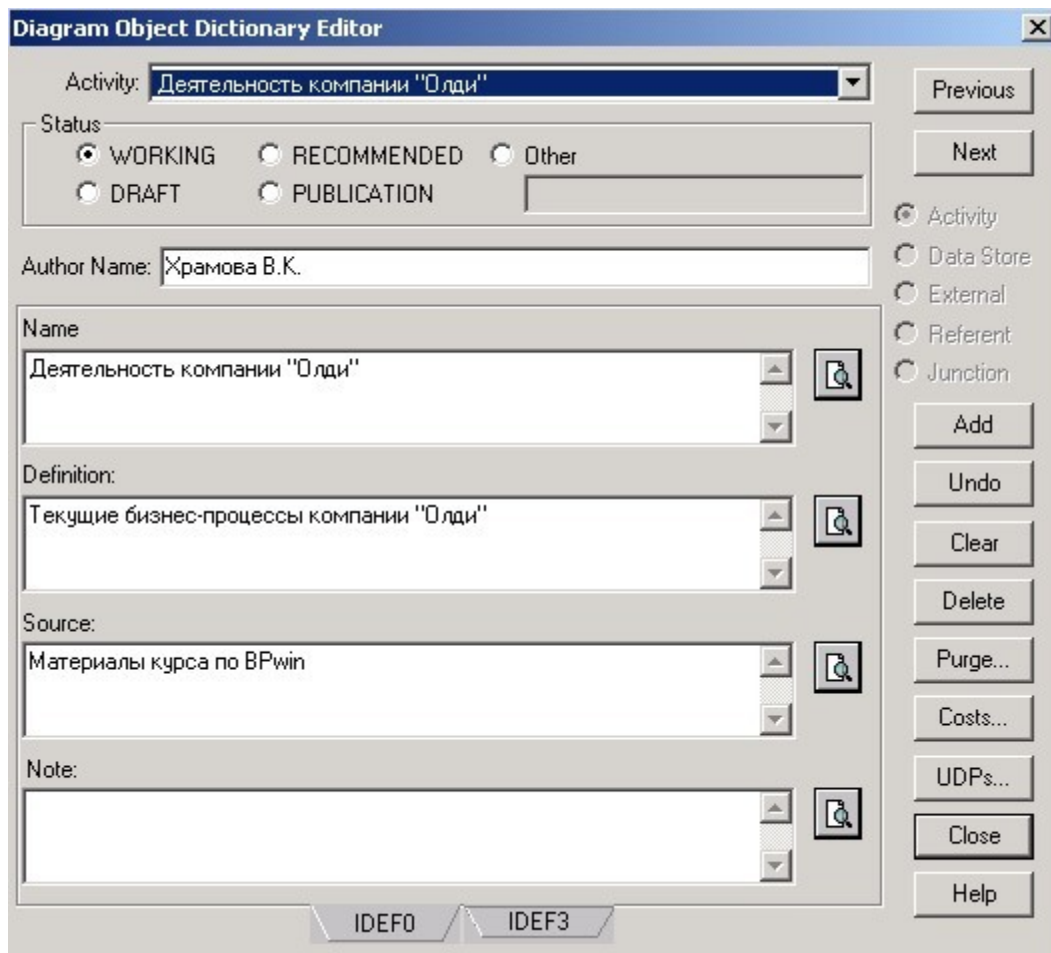




Рис.2.

3. Если Вы опишите имя и свойства работы в словаре, её можно будет внести в диаграмму позже с помощью кнопки  в палитре инструментов. Вы не можете удалить работу из словаря, если она используется на какой-либо диаграмме. Если вы удалите работу из диаграммы, из словаря она не удаляется. Имя и описание такой работы может быть использовано в дальнейшем. Для добавления работы в словарь щелкните по кнопке Clear, внесите имя и свойства работы, затем щелкните по Add. Для удаления всех работ, не используемых в модели, щелкните по Purge

3. Перейдите в режим рисования стрелок. Свяжите граничные стрелки (кнопка  на палитре инструментов) с остальными так как показано на рис.2. Для этого прочитайте раздел «Разветвляющиеся и сливающиеся стрелки» в Лабораторной работе №2

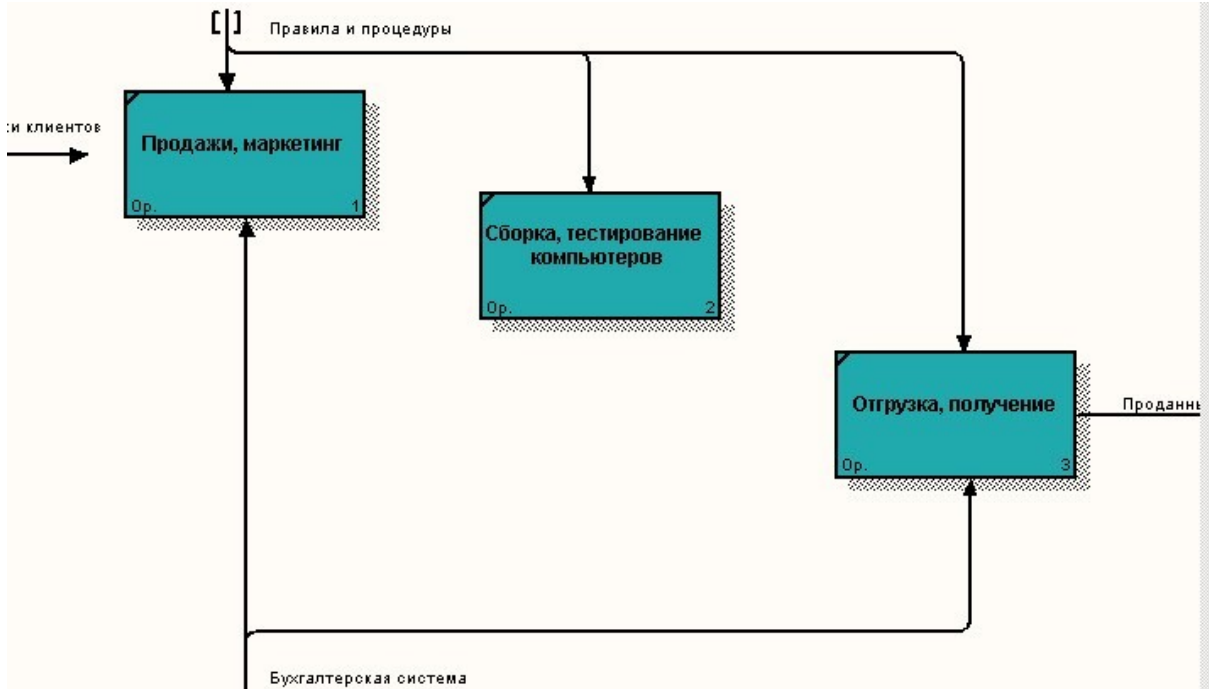


Рис. 3.

4. Правой кнопкой мыши щелкните по ветви стрелки управления работы «Сборка и тестирование компьютеров» и переименуйте её в «Правила сборки и тестирования» (рис.4.)

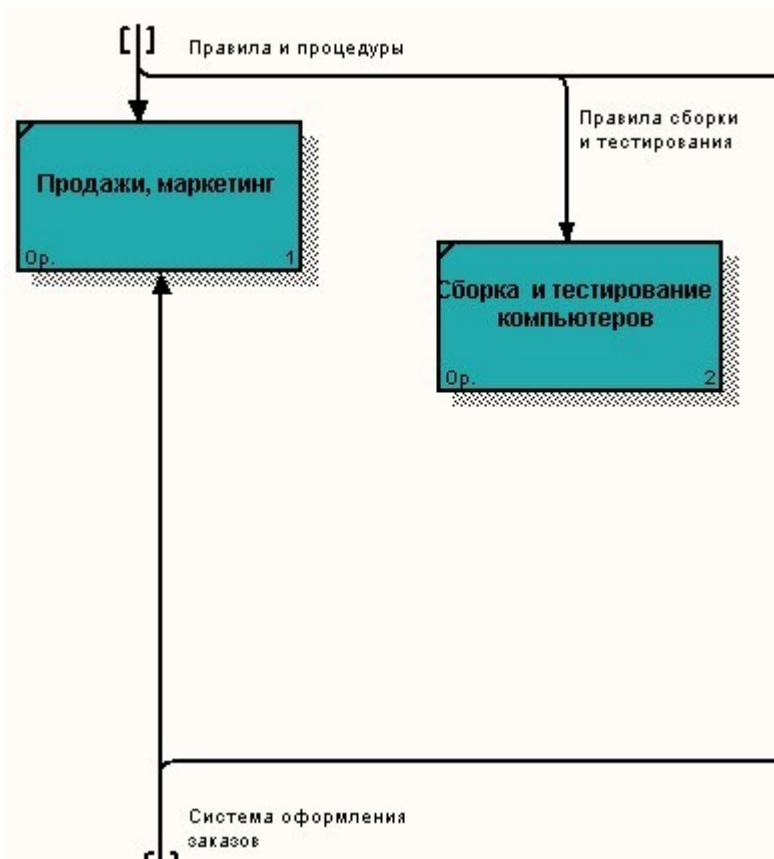


Рис. 4.

Внесите определение (Definition) для новой ветви: «Инструкция по сборке, процедуры тестирования, критерии производительности и т.д.»

Правой кнопкой мыши щелкните по ветви стрелки механизма работы «Продажи и маркетинг» и переименуйте её в «Систему оформления заказов»

5. Альтернативный метод внесения имен и свойств стрелок – использование словаря стрелок (вызов словаря – меню Model/Diagramm Object editor). Если вы опишите имя и свойство стрелки в словаре, её можно будет внести в диаграмму позже. Вы не можете удалить стрелку из словаря, если она используется на какойлибо диаграмме. Если Вы удалите стрелку из диаграммы, из словаря она не удаляется. Имя и описание такой стрелки может быть использовано в дальнейшем. Для добавления стрелки в словарь щелкните по кнопке Clear, внесите имя и свойства работы, затем щелкните по Add. Для удаления всех имен стрелок, не используемых в модели, щелкните по Purge Unused.
6. Создайте новые внутренние стрелки так, как показано на Рис.5.

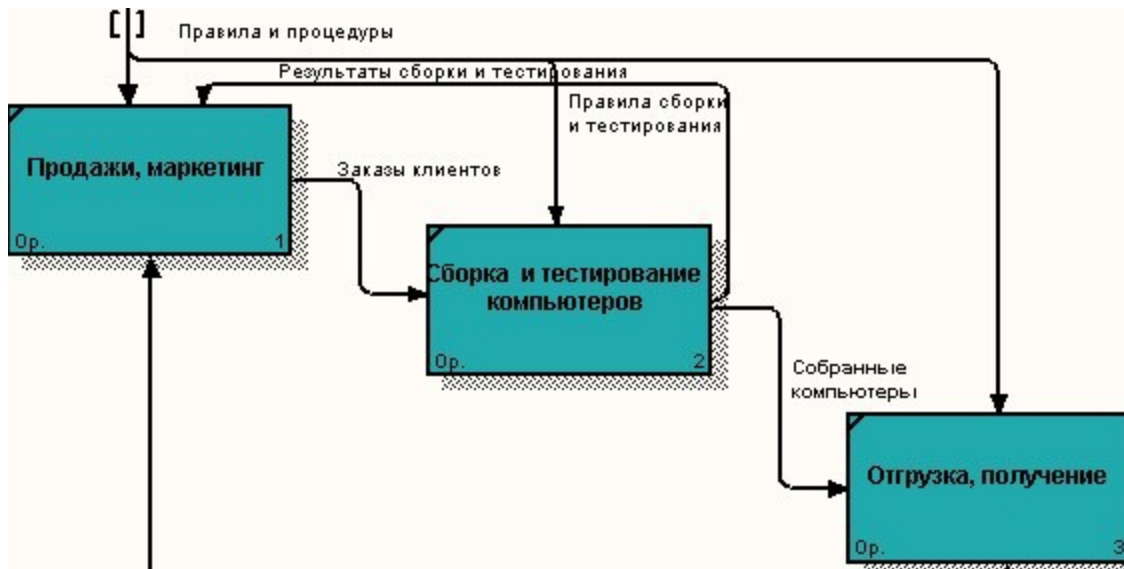


Рис.5.

7. Создайте стрелку обратной связи (по управлению) «Результаты сборки и тестирования» (рис.4.), идущую от работы «Сборка и тестирование компьютеров» к работе «Продажи и маркетинг». (В Л/Р №2 прочитайте материал: «Обратная связь по управлению»). Для большей наглядности измените стиль стрелки (толщина линий) и установите опцию Extra Arrowhead (из контекстного меню). Методом drag&drop перенесите имена стрелок так, чтобы их было удобнее читать. Если необходимо, установите Squiggle (из контекстного меню).
8. Создайте новую граничную стрелку выхода «Маркетинговые материалы» из работы «Продажи и маркетинг». Эта стрелка автоматически не попадает на диаграмму верхнего уровня и имеет квадратные скобки на кончике . Из палитры выберите кнопку , щелкните мышью по квадратным скобкам и в диалоговом окне Border Editor выберите Resolve Border Arrow. Для стрелки «Маркетинговые материалы» выберите опцию Trim из контекстного меню. Результат выполнения показан на рис.6. и в Приложении 1.



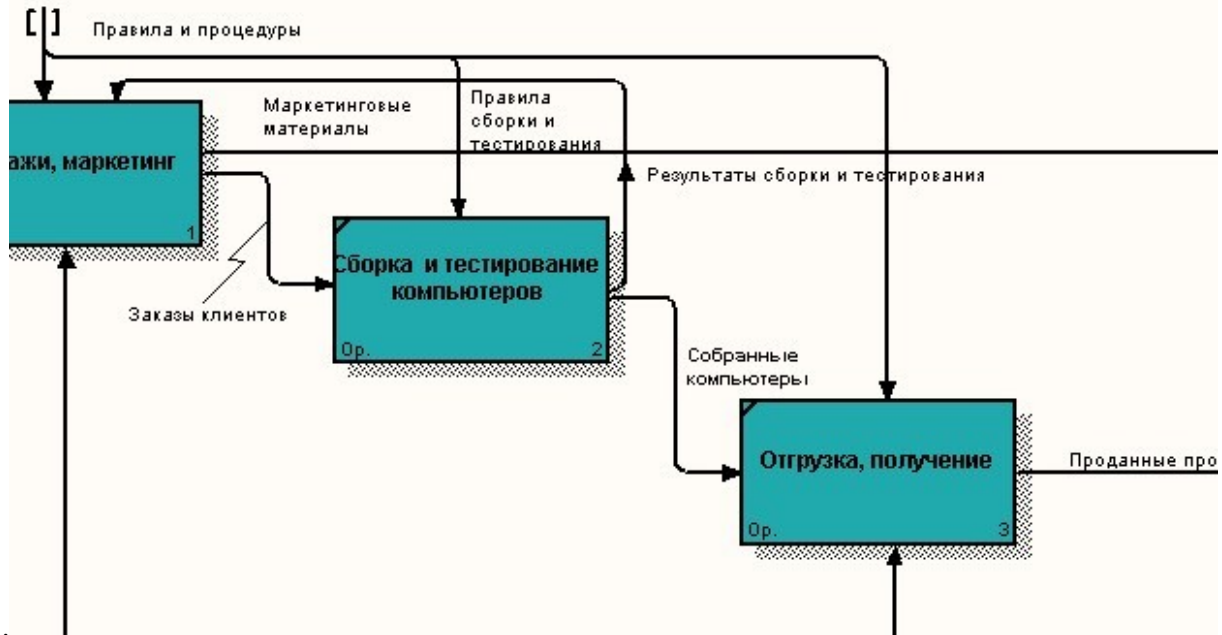


Рис.6.

Контрольные вопросы:

1. Как создать стрелку входа?
2. Как создать стрелку выхода ?
3. Как создать стрелку управления?
4. Как создать стрелку механизма?
5. Как посмотреть Словарь стрелок?
6. Как создать внутренние стрелки?
7. Как создать стрелки обратной связи?
8. Как создать разветвляющиеся и сливающиеся стрелки?
9. Как создать диаграмму декомпозиции?
10. Как создать отчет по модели?

## Создание диаграммы декомпозиции А.2

**Цель работы:**

Создание диаграммы декомпозиции для данного примера.

**Создание диаграммы декомпозиции.**

Декомпозируем работу «Сборка и тестирование компьютеров».

- В результате проведения экспертизы получена следующая информация. Производственный отдел получает заказы клиентов от отдела продаж по мере их поступления.
- Диспетчер координирует работу сборщиков, сортирует заказы, группирует их и дает указание на отгрузку компьютеров, когда они готовы.

- Каждые 2 часа диспетчер группирует заказы – отдельно для настольных компьютеров и ноутбуков – и направляет на участок сборки.
- Сотрудники участка сборки собирают компьютеры согласно спецификациям заказа по сборке. Когда группа компьютеров, соответствующая группе заказов, собрана, она направляется на тестирование. Тестировщики тестируют каждый компьютер и в случае необходимости заменяют неисправные компоненты.
- Тестировщики направляют результаты тестирования диспетчеру, который на основании этой информации принимает решение о передаче компьютеров, соответствующих группе заказов, на отгрузку.

1. На основе этой информации внесите новые работы и стрелки (табл.1. и табл.2.).

2.

Таблица 1. Работы диаграммы декомпозиции A2

Имя работы (Activity Name)	Определение работы (Activity Definition)
Отслеживание расписания и управление сборкой и тестированием	Просмотр заказов, установка расписания и выполнения заказов, просмотр результатов тестирования, формирование групп заказов на сборку и отгрузку
Сборка настольных компьютеров	Сборка настольных компьютеров в соответствии с инструкциями и указаниями диспетчера
Сборка ноутбуков	Сборка ноутбуков в соответствии с инструкциями и указаниями диспетчера
Тестирование компьютеров	Тестирование компьютеров и компонентов. Замена неработающих компонентов.

Таблица 2. Стрелки диаграммы декомпозиции A2

Имя стрелки (Arrow Name)	Источник стрелки (Arrow Source)	Тип источника стрелки (Arrow Source Type)	Назначение стрелки (Arrow Dest.)	Тип назначения стрелки (Arrow Dest. Type)

Диспетчер	Персонал производственного отдела		Отслеживание расписания и управление сборкой и тестированием	Механизм
Заказы клиентов	Граница диаграммы	Управление	Отслеживание расписания и управление сборкой и тестированием	Управление
Заказы на настольные компьютеры	Отслеживание расписания и управление сборкой и	Выход	Сборка настольных компьютеров	Управление

	тестированием			
Заказы на ноутбуки	Отслеживание расписания и управление сборкой и тестированием	Выход	Сборка ноутбуков	Управление
Компоненты	«Tunnel»	Вход	Сборка настольных компьютеров	Вход
			Сборка ноутбуков	Вход
			Тестирование компьютеров	Вход
Настольные компьютеры	Сборка настольных компьютеров	Выход	Тестирование компьютеров	Вход
Ноутбуки	Сборка ноутбуков	Выход	Тестирование компьютеров	Вход
Персонал производственного отдела	«Tunnel»	Механизм	Сборка настольных компьютеров	Механизм
			Сборка ноутбуков	Механизм
Правила сборки и тестирования	Граница диаграммы		Сборка настольных компьютеров	Управление

			Сборка ноутбуков	Управление
			Тестирование компьютеров	Управление
Результаты сборки и тестирования	Сборка настольных компьютеров	Выход	Граница диаграммы	Выход
	Сборка ноутбуков	Выход		
	Тестирование компьютеров	Выход		
Результаты тестирования	Тестирование компьютеров	Выход	Отслеживание расписания и управление сборкой и тестированием	Вход
Собранные компьютеры	Тестирование компьютеров	Выход	Граница диаграммы	Выход
Тестировщик	Персонал производственного отдела		Тестирование компьютеров	Механизм
Указание передать компьютеры на отгрузку	Отслеживание расписания управление и сборкой тестированием	Выход	Тестирование компьютеров	Управление

3. Тоннелируйте и свяжите на верхнем уровне граничные стрелки, если это необходимо.  
Результат выполнения работы показан в Приложении 1

Контрольные вопросы:

1. Как создать стрелку входа?
2. Как создать стрелку выхода ?
3. Как создать стрелку управления?
4. Как создать стрелку механизма?
5. Как посмотреть Словарь стрелок?
6. Как создать внутренние стрелки?
7. Как создать стрелки обратной связи?
8. Как создать разветвляющиеся и сливающиеся стрелки?
9. Как создать диаграмму декомпозиции?
10. Как создать отчет по модели?

**Критерии оценки работы:**

1. Полный ответ – 5 баллов.
2. Дополнительный уточняющий вопрос – 4 балла.
3. Краткий ответ – 3 балла.

## **САМОСТОЯТЕЛЬНАЯ РАБОТА № 2**

### **Создание диаграммы узлов. Создание диаграммы FEO. Каркас диаграммы**

**Цель работы:**

1. Изучение и создание диаграммы дерева узлов.
2. Изучение и создание диаграммы FEO.
3. Изучение содержания каркаса диаграммы

**Исходные данные (задание):**

**Задание №1.** Внимательно прочитайте «Создание диаграмм дерева узлов» **Создание диаграммы дерева узлов.**

Диаграмма дерева узлов показывает иерархию работ в модели и позволяет рассмотреть всю модель целиком, но не показывает взаимосвязи между работами(стрелки). Процесс создания модели работ является итерационным, следовательно, работы могут менять свое расположение в дереве узлов.

Для создания диаграммы дерева узлов следует выбрать в меню пункт

**Diagram/Add Node Tree.** Возникает эксперт создания диаграммы дерева узлов **Node Tree Wizard.** В первом диалоге эксперта необходимо внести имя диаграммы дерева узлов, узел верхнего уровня и глубину дерева **Number of Levels** (по умолчанию 3). Дерево узлов не обязательно в качестве верхнего уровня должно иметь контекстную работу и произвольную глубину. В одной модели можно создавать множество диаграмм деревьев узлов. Имя дерева узлов по умолчанию совпадает с именем работы верхнего уровня

Второй диалог эксперта **Node Tree Wizard** (рис.2.) позволяет задать свойства диаграммы дерева узлов.

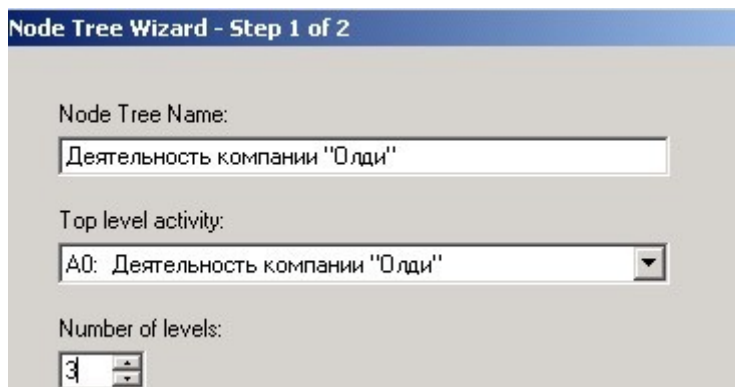


Рис.1. Первый диалог создания диаграммы дерева узлов

По умолчанию нижний уровень декомпозиции показывается в виде списка, а остальные работы в виде прямоугольников. Для отображения всего дерева в виде прямоугольников следует выбрать опцию **Bullet last level**. Группа **Connection Style** позволяет выбрать стиль соединительных линий.

Выберите меню **Diagram/Add Node Tree**. В первом диалоге вида **Node Tree Wizard** внесите имя диаграммы, укажите диаграмму корня дерева и количество уровней

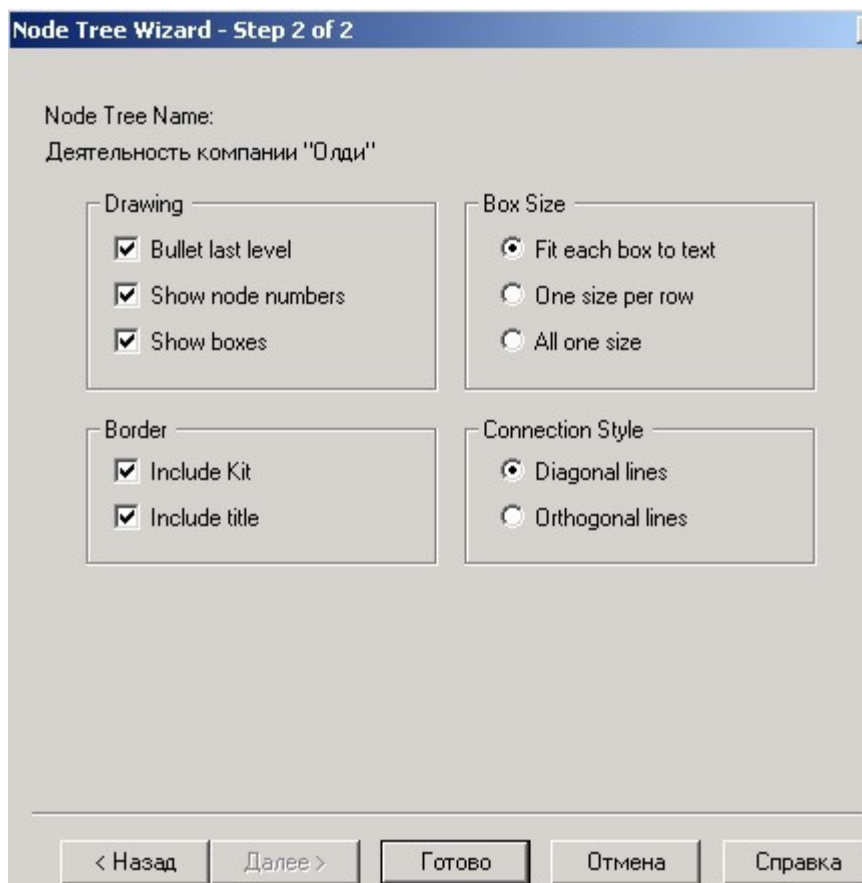


Рис.2. Диалог настройки диаграмм дерева узлов

Во втором диалоге установите опции, как на рис.2. Щелкните по кнопке «Готово». Создается диаграмма дерева узлов. Диаграмму дерева узлов можно модифицировать. Нижний уровень может быть отображен не в виде списка, а в виде прямоугольников, так же как и верхние уровни.

Для модификации диаграммы правой кнопкой мыши щелкните по свободному месту, не занятому объектами, выберите меню **Node Tree Diagram Properties** и во вкладке **Style** диалога **Node Tree Diagram Properties** отключите опцию **Bullet last level**. Щелкните по ОК.

1. Выберите пункт меню **Diagram/Add Node Tree**. В первом диалоге гида **Node Tree Wizard** внесите имя диаграммы, укажите диаграмму корня дерева и количество уровней.
2. Во втором диалоге установите опции, как показано на рис.2.
3. Щелкните по кнопке «Готово». Создается диаграмма дерева узлов. Диаграмму дерева узлов можно модифицировать. Нижний уровень может быть отображен не в виде списка, а в виде прямоугольников, так же как и верхние уровни.

Для модификации диаграммы правой кнопкой мыши щелкните по свободному месту, не занятому объектами, выберите меню **Node Tree Diagram Properties** и во вкладке **Style** диалога **Node Tree Diagram** отключите опцию **Bullet last level**. Щелкните по кнопке ОК.

**Задание №2.** Внимательно прочитайте «Создание FEO диаграммы»

### **Создание диаграммы FEO.**

Диаграммы «только для экспозиции» (FEO) часто используются в модели для иллюстрации других точек зрения, для отображения отдельных деталей, которые не поддерживаются явно синтаксисом IDEF0. Диаграммы FEO позволяют нарушить любое синтаксическое правило, поскольку по сути являются просто картинками – копиями стандартных диаграмм и не включаются в анализ синтаксиса. Например, работа на диаграмме FEO может не иметь стрелок управления и выхода. С целью обсуждения определенных аспектов модели с экспертом предметной области может быть создана диаграмма только с одной работой и одной стрелкой, поскольку стандартная диаграмма декомпозиции содержит множество деталей, не относящихся к теме обсуждения и дезориентирующих эксперта. Но если FEO используется для иллюстрации альтернативных точек зрения (альтернативный контекст), рекомендуется все-таки придерживаться синтаксиса IDEF0. Для создания диаграммы FEO следует выбрать пункт меню **Diagram/Add FEO Diagram**.

#### **Пример**

Предположим, что при обсуждении бизнес-процессов возникла необходимость детально рассмотреть взаимодействие работы «**Сборка и тестирование компьютеров**» с другими работами. Чтобы не портить диаграмму декомпозиции, создайте FEO- диаграмму, на которой будут только стрелки работы «**Сборка и тестирование компьютеров**».

1. Выберите пункт меню **Diagram/Add FEO Diagram**.
2. В диалоге **Add New FEO Diagram** выберите тип и внесите имя диаграммы FEO.
3. Для определения диаграммы перейдите в **Diagram/ Diagram Properties** и во вкладку **Diagram Text** внесите определение.

4. Удалите лишние стрелки на диаграмме FEO. Результат показан на рис.3.

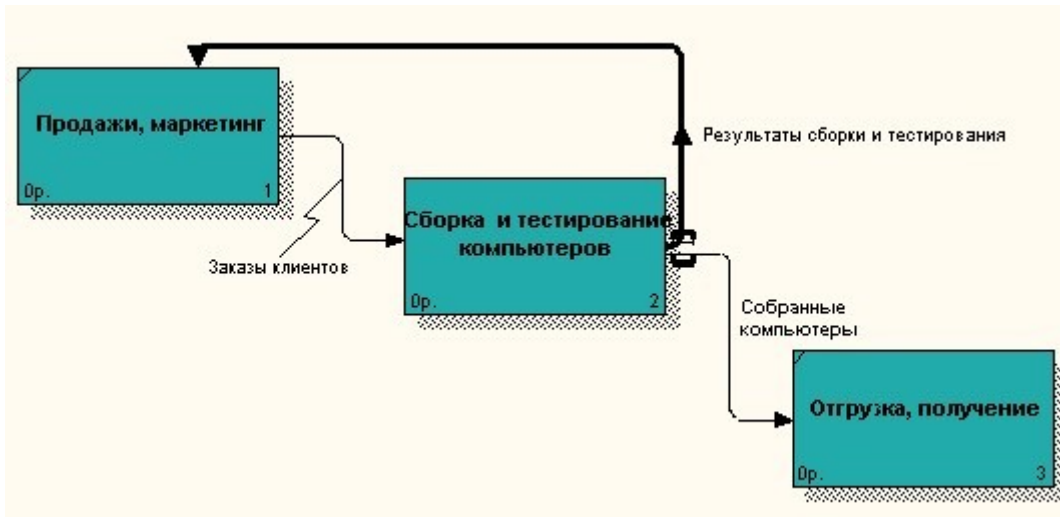



Рис.3. Диаграмма FEO

Для перехода между стандартной диаграммой, деревом узлов и FEO используйте кнопку  на панели инструментов.

#### Каркас диаграммы.

На рис.4 показан типичный пример контекстной диаграммы с граничными рамками, которые называются каркасом диаграммы. Каркас содержит заголовок (верхняя часть рамки, табл.1) и подвал (нижняя часть, табл.2). Заголовок каркаса используется для отслеживания диаграммы в процессе моделирования. Нижняя часть используется для идентификации и позиционирования в иерархии диаграмм. Значения полей каркаса задаются в диалоге Diagram Properties (в меню Edit/Diagram Properties).



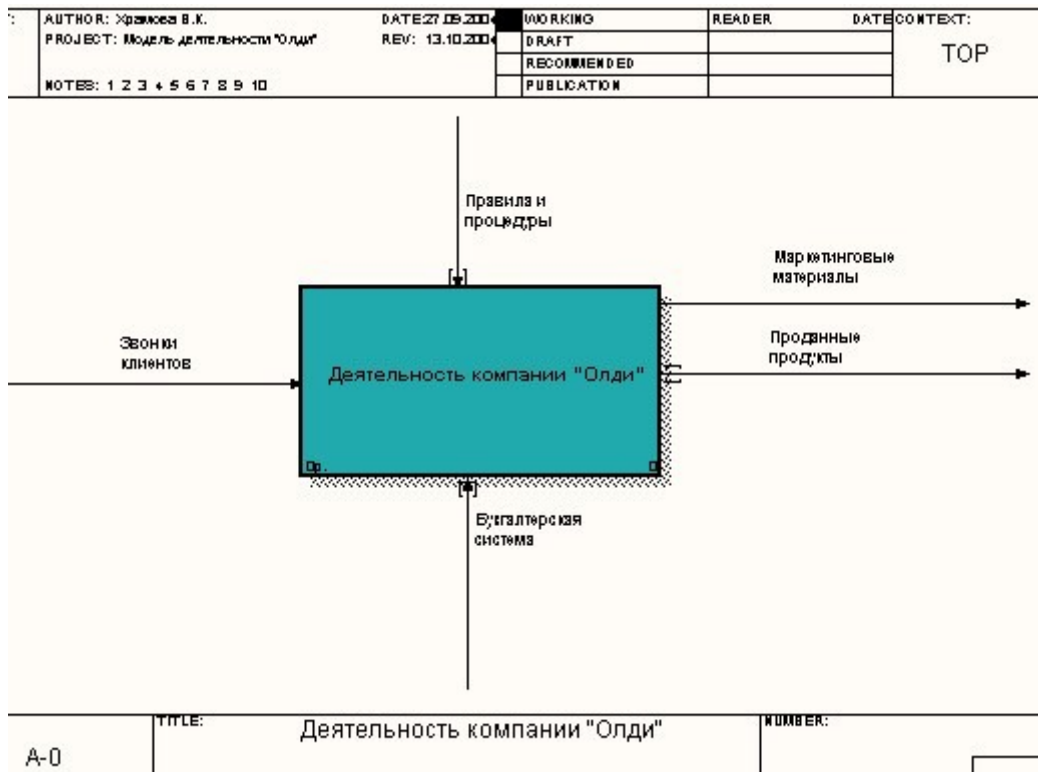


Рис.4. Контекстная диаграмма

### Поля заголовка каркаса (слева направо)

Таблица 1.

Поле	Смысл
<b>Used At</b>	Используется для указания на родительскую работу в случае, если на текущую диаграмму ссылались посредством стрелки вызова.
<b>Author, Date, Rev, Project</b>	Имя создателя диаграммы, дата создания и имя проекта, в рамках которого была создана диаграмма. REV - дата последнего редактирования диаграммы.
<b>Notes 1 2 3 4 5 6 7 8 9 10</b>	Используется при проведении сеанса экспертизы. Эксперт должен (на бумажной копии диаграммы) указать число замечаний, вычеркивая цифру из списка каждый раз при внесении нового замечания.
<b>Status</b>	Статус отображает стадию создания диаграммы, отображая все этапы публикации.
<b>Working</b>	Новая диаграмма, кардинально обновленная диаграмма или новый автор диаграммы.
<b>Draft</b>	Диаграмма прошла первичную экспертизу и готова к дальнейшему обсуждению.
<b>Recommended</b>	Диаграмма и все ее сопровождающие документы прошли экспертизу. Новых изменений не ожидается.

<b>Publication</b>	Диаграмма готова к окончательной печати и публикации.
<b>Reader</b>	Имя читателя (эксперта).
<b>Date</b>	Дата прочтения (экспертизы).
<b>Context</b>	Схема расположения работ в диаграмме верхнего уровня. Работа, являющаяся родительской, показана темным прямоугольником, остальные - светлым. На контекстной диаграмме (A-0) показывается надпись TOP. В левом нижнем углу показывается номер по узлу родительской диаграммы.

### **Поля подвала каркаса (слева направо)**

Таблица 2.

<b>Поле</b>	<b>Смысл</b>
<b>Node</b>	Номер узла диаграммы (номер родительской работы)
<b>Title</b>	Имя диаграммы. По умолчанию - имя родительской работы
<b>Number</b>	C-Number, уникальный номер версии диаграммы
<b>Page</b>	Номер страницы, может использоваться как номер страницы при формировании папки

#### Контрольные вопросы:

1. Как создать диаграмму дерева узлов?
2. Как создать FEO диаграмму?
3. Какую информацию содержит каркас диаграммы?
4. Какую информацию содержат поля подвала каркаса диаграммы?
5. В каком поле находится номер узла диаграммы?
6. В каком поле находится номер версии диаграммы?
7. В каком поле находится имя диаграммы?

#### **Критерии оценки работы:**

1. Полный ответ – 5 баллов.
2. Дополнительный уточняющий вопрос – 4 балла.
3. Краткий ответ – 3 балла.

## **САМОСТОЯТЕЛЬНАЯ РАБОТА № 3**

### **Создание отчетов в пакете VPwin**

### Цель работы:

Изучение способов создания отчетов в VPwin. **Исходные данные (задание):**

VPwin имеет мощный инструмент генерации отчетов. Отчеты по модели вызываются из пункта меню **Report**. Всего имеется семь типов отчетов:

1. **Model Report.** Этот отчет включает информацию о контексте модели - имя модели, точку зрения, область, цель, имя автора, дату создания и др.
2. **Diagram Report.** Отчет по конкретной диаграмме. Включает список объектов (работ, стрелок, хранилищ данных, внешних ссылок и т.д.).
3. **Diagram Object Report.** Наиболее полный отчет по модели. Может включать полный список объектов модели (работ, стрелок с указанием их типа и др.) и свойства, определяемые пользователем.
4. **Activity Cost Report.** Отчет о результатах стоимостного анализа.
5. **Arrow Report.** Отчет по стрелкам. Может содержать информацию из словаря стрелок, информацию о работе-источнике, работе-назначении стрелки и информацию о разветвлении и слиянии стрелок.
6. **Data Usage Report.** Отчет о результатах связывания модели процессов и модели данных.
7. **Model Consistency Report.** Отчет, содержащий список синтаксических ошибок модели.

Синтаксические ошибки IDEF0 с точки зрения VPwin разделяются на три типа:

- во-первых, это ошибки, которые VPwin выявить не в состоянии. VPwin не позволяет анализировать синтаксис естественного языка (английского и русского) и смысл имен объектов и поэтому игнорирует ошибки этого типа. Выявление таких ошибок - ручная работа.
- ошибки второго типа VPwin просто не допускает. Например, каждая грань работы предназначена для определенного типа стрелок. VPwin просто не позволит создать на диаграмме IDEF0 внутреннюю стрелку, выходящую из левой грани работы и входящую в правую грань.
- третий тип ошибок VPwin позволяет допустить, но отмечает их. Полный их список можно получить в отчете Model Consistency Report. Это единственный неопциональный отчет в VPwin. Список ошибок может содержать, например, неименованные работы и стрелки (unnamed arrow, unnamed activity), несвязанные стрелки (unconnected border arrow), неразрешенные стрелки (unresolved (square tunneled) arrow connections), работы, не имеющие по крайней мере одной стрелки выхода и одной стрелки управления, и т.д.

При выборе пункта меню, который соответствует какому-либо отчету, появляется диалог настройки отчета. Для каждого из семи типов отчетов он выглядит посвоему. Рассмотрим типичный диалог Arrow Report (рис.7).

Раскрывающийся список Standard Reports позволяет выбрать один из стандартных отчетов. Стандартный отчет - это запоминаемая комбинация переключателей, флажков и других элементов управления диалога. Для создания собственного стандартного отчета необходимо задать опции отчета, вве-

сти имя отчета в поле списка выбора и щелкнуть по кнопке New. BPwin сохраняет информацию о стандартном отчете в файле BPWINRPT.INI. Все определения этого файла доступны из любой модели. Единственное ограничение - свойства, определяемые пользователем (User Defined Properties). Они сохраняются в виде указателя и поэтому доступны только из родной модели. Стандартный отчет можно изменить или удалить.

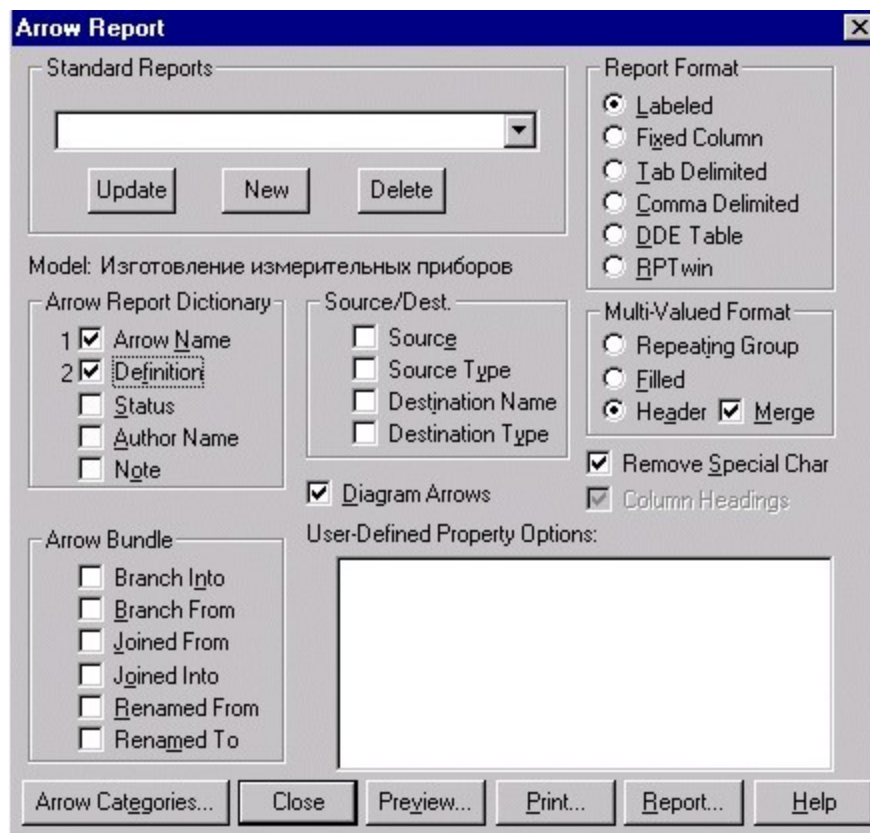


Рис.7.Диалог настройки отчета

В правом верхнем углу диалога находится группа управляющих элементов для выбора формата отчета. Доступны следующие форматы:

- **Labeled** - отчеты включают метку поля, затем, в следующей строке, печатается содержимое поля;
- **Fixed Column** - каждое поле печатается в собственной колонке;
- **Tab-Comma Delimited** - каждое поле печатается в собственной колонке. Колонки разделяются знаком табуляции или запятыми;
- **DDE Table** - данные передаются по DDE приложению, например MS Word или Excel;
- **RPTwin** - отчет создается в формате Platinum RPTwin - специализированного генератора отчетов, который входит в поставку BPwin.

Опция **Ordering** (на отчете по стрелкам отсутствует) сортирует данные по какомулибо значению. Опция Multi-Valued Format регулирует вывод полей в отчете при группировке данных:

- **Repeating Group** - детальные данные объединяются в одно поле, между значениями вставляется +.

- **Filled** - дублирование данных для каждого заголовка группы;
- **Header** (опция по умолчанию) - печатается заголовок группы, затем - детальная информация.

Задание №1. Создать отчет о контексте модели, используя различные опции.

Задание №2. Создать отчеты по конкретным диаграммам.

Задание №3. Создать полный отчет по модели.

Задание №4. Создать отчет о результатах стоимостного анализа.

Задание №5. Создать отчет по стрелкам.

Задание №6. Создать отчет, содержащий список синтаксических ошибок модели.

Контрольные вопросы:

1. Как создать отчет о контексте модели?
2. Как создать отчеты по конкретным диаграммам?
3. Как создать полный отчет по модели?
4. Как создать отчет о результатах стоимостного анализа?
5. Как создать отчет по стрелкам?
6. Как создать отчет, содержащий список синтаксических ошибок модели?

*Вопросы для итогового контроля:*

1. Понятие «инструментальное средство разработки программного обеспечения». Типы таких инструментальных средств.
2. Трансляторы. Классификация. Назначение. Примеры.
3. Возможности транслятора gcc.
4. Интегрированные среды разработки. Классификации. Назначение. Примеры. Особенности шести из следующих инструментальных средств: WEB Storm, JDeveloper, NetBeans, IntelliJ IDEA, Eclipse, KDevelop, MonoDevelop, QT Creator, XCode, PyCharm, Rational Application Developer, Android Studio.
5. Инструменты автоматизации сборки. Назначение. Примеры. Особенности шести из следующих инструментальных средств: Apache ANT, CMake, Jenkins, MSBuild, Nant, distcc, cabal, automake, autotools, scons, Apache Maven, Waf, Rake.
6. Инструмент автоматизации сборки make.

7. Инструменты автоматизации сборки automake/autoconf.
8. Назначение и использование инструментов учета обращений (issues).
9. Методики создания статической и динамической библиотек в среде POSIX.
10. Инструмент анализа покрытия кода gcov
11. Средства контроля версий. Классификация. Назначение. Примеры. Особенности шести из следующих инструментальных средств: ClearCase, CVS, Darcs, Revision Control System, Subversion, Visual Source Safe, PVCS, BitKeeper, Gnu Bazaar, Team Foundation Server, Rational Team Concert, Mercurial, Code co-op.
12. Средство контроля версий git.
13. Средства измерения производительности. Назначение. Примеры. Особенности шести из следующих инструментальных средств: Valgrind, Alinea MAP, DevPartner, LTTng, OProfile, VTune, NetBeans Profiler, CodeAnalyst, Firebug, AQtime, Intel Parallel Studio, Visual Studio
14. Инструмент измерения производительности и поиска утечек памяти valgrind.
15. Инструмент автоматизации тестирования AutoIT.
16. UNIT-тестирование. Инструментальные средства для UNIT-тестирования. Пример.
17. Статический анализ кода. Инструментальные средства статического анализа кода. Примеры. Особенности шести из следующих инструментальных средств: cppcheck, dafny, coverity, kiuwan, LDRA, Malpas, Polyspace, Klockwork, SemmlCode, продукты AdaCore, VeraCode, CodeRush Classic, CODAN в Eclipse, PVS-Studio, Goanna, продукты PRQA, средства IntelliJ IDEA, Astr'ee
18. Инструментальные средства управления требованиями (на примере, выбранном студентом).
19. DOCBOOK. Краткое описание способа использования. Достоинства и недостатки.
20. Технология wiki.
21. Особенности шести из следующих средств/способов подготовки документации: texinfo, docutils, Dr. Explain, wiki, StepShot, Help&Manual, Makrdown, AuthorIT, ClickHelp, FrameMaker, MadCap Flare, Doc-to-Help, Help Generator, StepShot, HelpStudio, FastHelp, Doc-O-Magic, Helpinator, HelpSmith, Softany Software, Daux.io, Latex
22. Инструментальные средства отслеживания ошибок. Назначение. Возможности. Примеры.
23. Инструментальные средства проектирования. Назначение. Возможности. Примеры. Особенности шести из следующих инструментальных средств: Rational Rhapsody, Astah, Papyrus, StarUML, Software Ideas Modeler, UML Designer, Enterprise Architect, Edraw Max, Modelio, Glify, yEd, BOUML, MagicDraw, PlantUML, LucidChart, Real Time Developer Studio, Umbrello, UMLet, Prosa UML Modeller, Visual Paradigm for UML, Rational Software Architect, UModel, CaseComplete, Rational System Architect, NetBeans, Microsoft Visio.

24. Средства обратной разработки. Назначение. Возможности. Пример.
25. Примеры средств автоматического и автоматизированного программирования.
26. Средство автоматизированного программирования LEX. Назначение. Возможности.
27. Инструментальные средства, используемые на базе практики
28. Выбор инструментальных средств в проекте (на примере)