

Технология программирования

Курс лекций для групп 3057, 4057

Лекция №1

Лектор: проф. И.В. Штурц

2010 г.

Организационные вопросы

- 10 лекций, экзамен
- Практика (по сути, курсовой проект) - зачет
 - у групп 3057 (все) и 4057/2
- Презентации лекций — на www.d-inter.ru/SE
 - не отменяют чтения литературы
- Оценка на экзамене складывается из баллов за:
 - посещение лекций
 - сдачу зачета в срок и качественно (у кого есть зачет)
 - ответ на экзамене
 - как минимум, знание всех основных понятий
 - проверяется знакомство с литературой
 - задаются дополнительные вопросы (публикуются в конце презентации лекции)
- Лектор: Игорь Викторович Штурц
ishturts@gmail.com
921-998-9222

Название курса лекций

Software Engineering; варианты перевода:

- Программная инженерия
- Инженерия программного обеспечения
- Софтостроение
- Технология промышленной разработки программных продуктов
- Общепринятый эквивалент:

Технология программирования

Предмет курса лекций

До сих пор вы изучали «программирование в малом»: разработку алгоритмов и отдельных небольших программ по готовым техническим заданиям (специализация - кодировщик)

Предмет данного курса: «программирование в большом», т.е. все виды профессиональной деятельности программистов, выходящие за пределы программирования в малом:

- анализ требований
- составление технических заданий
- проектирование больших программных систем
- верификация и аттестация (валидация)
- тестирование
- написание документации
- внедрение и сопровождение программных продуктов
- организация команды разработчиков
- словесная коммуникация участников разработки
- управление проектами

Специализация – software engineer, инженер по разработке ПО


Программирование в большом vs. в малом

- Больше объектов в поле зрения одновременно
- Работа в составе коллектива (команды, team)
 - Сопряжение частей - согласование их интерфейсов
 - Коммуникация – передача знания
 - Социально-психологическая гармония
- Языки, отличные от языков математики и программирования:
 - языки описания проекта
 - естественные языки: русский, английский
- Неопределенность и изменчивость требований

«Программирование в большом» занимает 80-85% общей трудоемкости создания программных продуктов, «в малом» -15-20%

Для успешной карьеры инженеру ПО нужны не только математические/технические, но и гуманитарные знания и умения

Содержание курса лекций

1. Введение
 - Обзор курса лекций
 - Основные понятия технологии программирования
 - Виды программных продуктов (ПП)
 - Критерии качества ПП
 2. Модели жизненного цикла ПП
 3. Модели команды разработчиков
 4. Управление проектами
 5. Документирование и словесная коммуникация
 6. Структуризация ПП: модульное и компонентное программирование
 7. Языки и методы проектирования
 8. Тестирование и верификация ПП
 9. CASE-системы
 10. Надежность ПП, ее оценка и меры по ее повышению
 11. Стандарты качества технологии программирования
- 
- Сегодня

Что вы узнаете из курса лекций

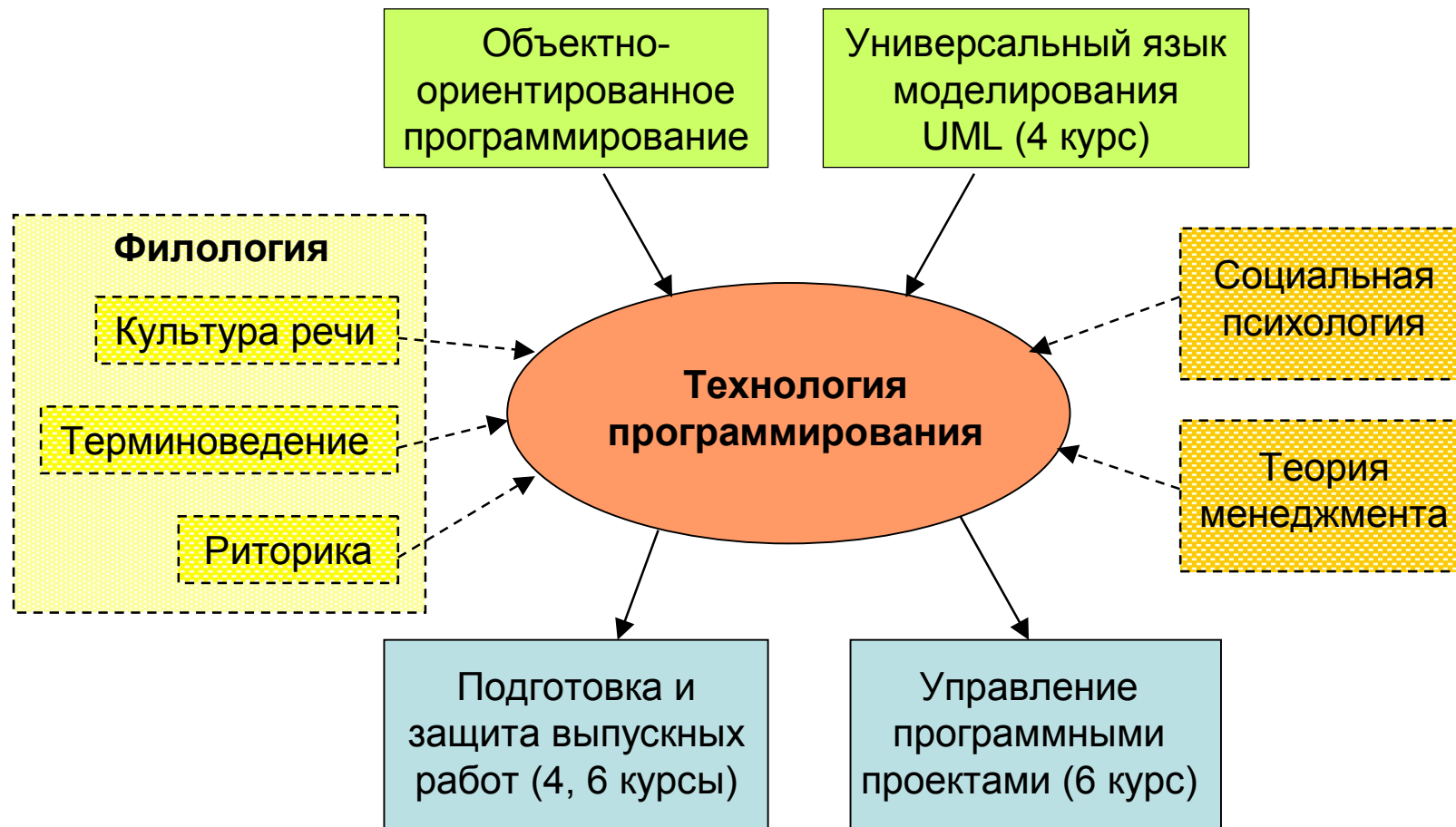
- Как должен быть организован производственный процесс разработки ПП:
 - последовательность этапов во времени и их взаимосвязь
 - распределение обязанностей в команде
 - планирование и контроль
- Как писать «техническую прозу» - программную документацию
- Как эффективно общаться с сотрудниками и заказчиками
- Что такое сборочное и компонентное программирование
- На каких языках описывается проект ПП
- Как осуществляется целенаправленное тестирование ПП
- Что такое формальная верификация программ
- Как измеряют надежность ПП и повышают ее
- Каков состав и функции CASE-средств
- Что требуют международные стандарты качества технологии

Практические занятия (зачет)

Разработать небольшой пакет программ:

- Проектная документация
- Код на объектно-ориентированном языке
- Тестовый набор

Связь с другими учебными дисциплинами



Computer Science vs. Software Engineering

Прикладная математика vs. Технология программирования

Раздел математики

Инженерная наука

Развивает новые
формальные методы

Применяет известные
теоретические методы

Цель – методология

Цель – промышленный продукт

Предмет – алгоритмы

Предмет – программы и средства
их создания

Литература (1)

Основная

1. С.Макконнелл. Совершенный код, 2-е изд. Майкрософт пресс, 2005.
2. Ф.Брукс. Мифический человеко-месяц или как создаются программные системы. - М., Мир, 1988; СПб, «Питер», 2000.
3. И.Соммервилл. Инженерия программного обеспечения. - «Вильямс», 2002.
4. С.А.Орлов. Технологии разработки программного обеспечения. - СПб, «Питер», 2003.

Дополнительная

1. Э.Йордан. Путь камикадзе. Как разработчику программного обеспечения выжить в безнадежном проекте. www.books-shop.com, 2002.
2. Л.Константин. Человеческий фактор в программировании. – СПб – М., Символ, 2002.
3. Д.Мараско. IT проекты: фронтовые очерки. «Символ», СПб – М., 2008.

Литература (2)

Электронные курсы лекций Университета информационных технологий

1. Д.В. Кознов, Д.Ю. Бугайченко. Введение в программную инженерию
<http://www.intuit.ru/department/se/inprogeng/>
2. В.А. Петрухин, Е.М. Лаврищева. Методы и средства инженерии программного обеспечения
<http://www.intuit.ru/department/se/swebok/>
3. И.Н. Скопин. Основы менеджмента программных проектов
<http://www.intuit.ru/department/se/msd/>
4. С.В. Синицын, Н.Ю. Налютин Верификация программного обеспечения
<http://www.intuit.ru/department/se/verify/>
5. А.Н. Терехов Введение в технологию программирования
<http://www.intuit.ru/department/se/introprogteach/>
6. В.И. Грекул Проектирование информационных систем
<http://www.intuit.ru/department/se/devis/>

Что из литературы знать к экзамену в первую очередь

- С.Макконнелл. Совершенный код: части 2, 5 и 7
- Д.Мараско. IT проекты: фронтовые очерки: части 2 и 3 (в частности, 9 главу)
- С. Синицын, Н.Налютин. Верификация программного обеспечения: лекции 3-6, 12.

Основные понятия

- Программный продукт (ПП) (software); синонимы: *программное обеспечение (ПО)*, *программное изделие* (ГОСТ 1988 г.)
 - software vs. program = товар vs. код

Промышленный продукт (товар):

- распространяется = отчуждается от производителя, часто также тиражируется
- имеет документацию и гарантии качества = гарантийное обслуживание

Программный продукт - документированная программа с гарантированным уровнем качества, а часто и с гарантийным обслуживанием (сопровождением)

Промышленное производство vs. кустарное или научные исследования

Серийное и массовое производство vs. индивидуальное:

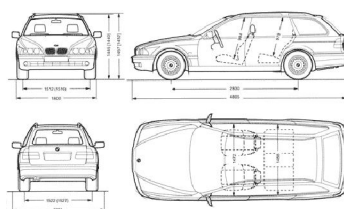
- стандартные компоненты (детали), инструменты и приемы
- коллективный труд → разделение труда и специализация (конвейер):
 - системный аналитик
 - архитектор
 - проектировщик/ кодировщик
 - тестировщик
 - технический писатель
 - руководитель проекта
 - и др.
- автоматизация производства: CASE–системы (Computer Aided Software Engineering)

Программная инженерия vs. традиционная инженерия

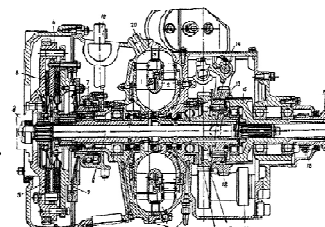
Дизайн



Проект



Чертежи

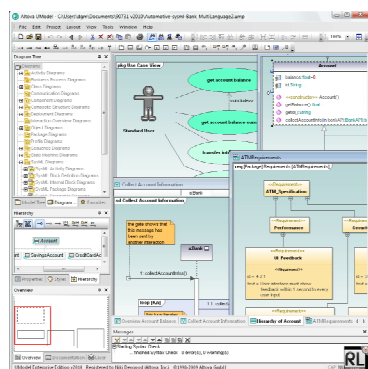


Производство



Использование

Проектирование → Конструирование → Изготовление



Архитектура

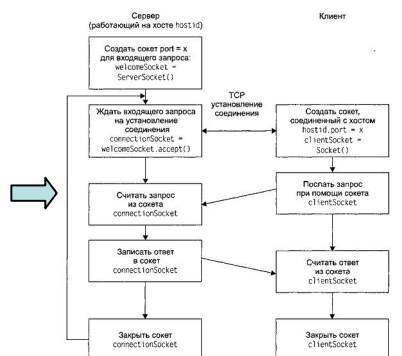


Рис. 2.20. Приложение клиент/сервер, использующее транспортные службы с установлением логического соединения

Детальный проект

```

1  // Сервер - [tcp_listen] - Gray
2  //
3  // * tcp_listen.c - программа осуществляет обмен данными с клиентом
4  // *
5  // *
6  // *
7  // *
8  // *
9  // *
10 // *
11 // *
12 // *
13 // *
14 // *
15 // *
16 // *
17 // *
18 // *
19 // *
20 // *
21 // *
22 // *
23 // *
24 // *
25 // *
26 // *
27 // *
28 // *
29 // *
30 // *
31 // *
32 // *
33 // *
34 // *
35 // *
36 // *
37 // *
38 // *
39 // *
40 // *
41 // *
42 // *
43 // *
44 // *
45 // *
46 // *
47 // *
48 // *
49 // *
50 // *
51 // *
52 // *
53 // *
54 // *
55 // *
56 // *
57 // *
58 // *
59 // *
60 // *
61 // *
62 // *
63 // *
64 // *
65 // *
66 // *
67 // *
68 // *
69 // *
70 // *
71 // *
72 // *
73 // *
74 // *
75 // *
76 // *
77 // *
78 // *
79 // *
80 // *
81 // *
82 // *
83 // *
84 // *
85 // *
86 // *
87 // *
88 // *
89 // *
90 // *
91 // *
92 // *
93 // *
94 // *
95 // *
96 // *
97 // *
98 // *
99 // *
100 // *
    
```

Код

Тиражирование и использование

В программном продукте велика доля интеллектуальной составляющей

8 фев 2010

15

Понятие технологии программирования

- **Технологический процесс** - последовательность действий (технологических операций), направленных на создание продукта
- **Технология** изучает и определяет технологические процессы и порядок их выполнения (стадии) с использованием определенных знаний, методов и средств
Применительно к технологии программирования (ТП):
 - **Знания** математических методов, алгоритмов и языков прог-я
 - **Методы** разработки (проектирования, кодирования, тестирования, отладки, организации коллективного труда, коммуникации и управления проектами)
 - **Средства** - системы программирования, инструментальные программные средства (tools, toolkits, CASE-средства), аппаратура (компьютеры, dev kits, сети, ...)
- Более узкое понятие – **методология программирования** – это теоретическая база ТП: знания + методы

Каждый софтверный производитель имеет свою документированную ТП
Распространяемые ТП: MSF (Microsoft Solutions Framework), IBM RUP, ...

Тонкости терминологии

- Англ. термин «Technology» в словаре Lingvo:
 1. Техника, технические науки
 2. Технология
- Термин «Информационные технологии» (ИТ) - неточная калька с Information Technology
 - Точнее перевод «Информационная техника»
 - Раньше это понятие называлось «Вычислительная техника и техника связи»
- Технология программирования = software engineering (не technology!)

Таким образом, в русском словосочетании «информационные технологии» «технология» обозначает прежде всего предмет, а в словосочетании «технология программирования» - процесс, способ

Перманентный «кризис программирования»

- впервые объявлен в 1968 г.

Программирование – это отрасль высоких технологий, перешедшая из сферы научных исследований в промышленное производство

Особенности:

- Велика доля интеллектуальной составляющей в конечном продукте
- Постоянно растущие размеры и сложность ПП
- Интенсивная коммуникация между участниками - передача знания в процессе производства
- Велика доля неудачных программных проектов. По статистике в США – это в среднем 3 из каждых 4-х:
 - 25% *проваленных* – проекты не были доведены до конца
 - 50% *проблемных* – проекты не уложились в сроки, перерасходовали бюджет и/или было сделано не все, что требовалось
 - Только 25% *успешных* проектов – вовремя и в рамках бюджета был выполнен весь намеченный объем работ

(Статистика по 30 тыс. программных проектов в США за 1995-2001, собранная Microsoft Consulting Services)

Основные причины неудач

- Нечёткие или неполные требования
 - Постоянное изменение требований
 - Слишком широко поставленная задача
 - Нечётко сформулированные цели
 - Низкое качество кода
 - Ошибки прогноза трудоемкости и планирования
 - Ошибки в подборе кадров
 - Плохая организация работы
- т.е., в основном организационные, а не технические причины:
работа с **peopleware**

Большая сложность ПП мешает эффективному
планированию и управлению проектами

Характеристики программных проектов

Размер, строк исх. кода	Число разработ- чиков	Время разра- ботки	Трудо- емкость, чел-лет	% прова- ленных проектов	% проектов с нарушениями	
					сроков	бюджета
10^3	1	1 мес.	10^{-1}	-	-	-
10^4	1-2	6 мес.	1	-	15%	-
10^5	10	1 год	10^1	15%	20%	25%
10^6	100	1 – 5 лет	10^2	50%	90%	85%
10^7	>100	2 – 8 лет	$10^3 – 10^4$	65%	-	-
11 млн	200	7 лет	1400	Win 95	← Примеры больших проектов	
40 млн	5700	2 года	~10000	Win XP		
5 млн	400	3 года	1200	Боинг 777		

Два главных вызова ТП

Проблема №1: постоянный рост выч. мощностей → рост потребности в программах опережает рост числа программистов

Выход: повышать производительность труда в отрасли софтостроения
Производительность - количество строк исходного кода в день - почти не растет (в среднем 10-50 строк исходного кода на весь цикл проектирование-отладка)

Решения – крупноблочное софтостроительство:

- 1) разрабатывать универсальные настраиваемые (customized) ПП:
 - настройка параметрическая и процедурная (на скриптовом языке)
напр., АСУ (ERP): SAP, Ахapta, 1С, ...
- 2) собирать программы из готовых крупных частей - полуфабрикатов:
сборочное программирование и повторное использование компонентов (reusability)
- 3) автоматизировать рутинные производственные процессы: CASE

Два главных вызова ТП (2)

Проблема №2: все больше вычислит. систем критической важности (mission-critical) с повышенными требованиями к надежности



Решения для повышения надежности ПП :

- 1) Количественные оценки и измерения надежности программ
- 2) Предупреждение ошибок при проектировании и кодировании
- 2) Интенсивное целенаправленное тестирование
- 3) Формальная верификация ПП – доказательство их правильности
- 4) Построение отказоустойчивых систем

Виды программных продуктов

Дихотомийная классификация

Заказной (custom) - для конкретного заказчика, малый (часто единичный) тираж



Рыночный (commercial, public) - для широкого круга пользователей; неограниченный тираж

Встроенный (embedded) – для управляющего (микро)компьютера; поставляется в составе системы управления (в ПЗУ или ОЗУ)



Загружаемый (loadable) - может загружаться и выполняться на любой машине данной платформы

Библиотека (library) - полу-фабрикат для программистов



Приложение (application) – адресовано конечным пользователям

Прикладной (application) - специализированный для задач конечных пользователей



Системный (system) – универсальный: изолирующий приложения от аппаратуры (как ОС) или служащий инструментом (tool) (как компилятор)

Отступление об относительности любой классификации в мире вещей

- «Промежуточные» виды ПП:
 - архиватор RAR: системная или прикладная программа?
 - DLL-плагин: библиотека или приложение?
- Почти всегда находятся объекты, по некоторым признакам относящиеся к одному классу, а по другим – к другому
 - в физике: электрон – и волна, и частица
 - в биологии: неоднозначность классификации организмов по Линнею
- Это потому, что любая классификация – абстракция в мире идей, проецируемая на мир вещей, который не обязан ей подчиняться
- Только в математике возможна абсолютно однозначная классификация объектов, поскольку вся математика – абстракция, т.е. принадлежит миру идей
 - напр., вещественные числа либо рациональные, либо иррациональные – третьего не дано

Поэтому неправильно говорить, что в некоторой предметной области
«еще не сложилась четкая классификация»

Классификация объектов в предметном мире всегда нечеткая!

Классификация ПП по назначению

т.е. по преобладающему виду обрабатываемой информации

А. Числа – Математическое моделирование процессов в природе, технике и обществе

- Основа - численные методы
- Интегрируются с компьютерной графикой для визуализации ввода-вывода (САПР, научная визуализация)

В. Тексты и числа - Системы обработки данных (Data processing)

От простых текстовых редакторов до экономических информационных систем (АСУ или ERP - Enterprise Resource Planning)

С. Сигналы и изображения

- Системы реального времени - в них вычислительный процесс синхронизируется внешними событиями и время реакции ограничено сверху:
 - встроенные системы управления (embedded systems)
 - АСУТП (АСУ технологическими процессами)
 - цифровая связь и пр.
- Мультимедиа (обучающие программы, игры, музыка, речь, видео, ...)
- Web-приложения
- Полиграфия, картография (GIS – Geographical Information Systems)
- и пр.

Характеристики качества ПП

А. Потребительские - точка зрения заказчика и пользователя

1. **Функциональность** (functionality)
2. **Удобство** использования и эксплуатации (usability) – дружелюбный UI, эксплуатационная документация
3. **Надежность** (reliability):
 - ❖ Безотказность = безошибочность (correctness)
 - ❖ Отказоустойчивость (fault-tolerance, resilience) - способность продолжать работу после отказа
4. **Эффективность** (efficiency) – минимум требуемого времени, памяти и других ресурсов
5. **Совместимость** (compatibility):
 - либо взаимозаменяемость (преемственность свойств)
 - Пример: новая версия компилятора транслирует старые программы
 - либо стыкуемость по вв/вы, т.е. по форматам файлов
 - Пример: ACDSee совместим с Photoshop по форматам файлов растровой графики
6. **Сопрягаемость** (interoperability) - возможность совместной работы и взаимодействия с определенными другими программами
7. **Масштабируемость** (scalability) - возможность настройки/расширения для работы на аппаратных конфигурациях различных размеров/мощности
8. **Безопасность** (security) – защищенность от несанкционированных действий (отсутствие уязвимостей и защита)

Характеристики качества ПП (2)

Б. Технологические и рыночные - точка зрения производителя

- **Понятность** (readability), модульность, наличие проектной документации – все вместе называют сопровождаемостью (maintainability)
- **Многоплатформенность** (multi-platform) – возможность выполнения на разнообразных платформах
- **Переносимость** (portability, mobility) - простота переноса на другие платформы
- **Расширяемость** (extendability) – простота добавления новой функциональности
- Возможность **повторного использования** (reusability)
- **Защищенность** от несанкционированного копирования (security)

Отступление об упорядочении и группировании текстовых списков

- Списки – перечни однородных сущностей - следует упорядочивать по наиболее подходящему признаку, если такой есть:
 - по важности (приоритету)
 - по размеру, весу, стоимости, ...
 - по времени, если это события
 - по алфавиту, если это имена собственные
 - и т.п.
- Упорядочение подчеркивается нумерацией строк списка
- Если элементы списка полностью равноправны (т.е., отсутствует признак для упорядочения) то нумерация не нужна – используйте маркированный (bulleted) список
- Разбивайте длинные списки на группы длиной не более 7 – 10 элементов каждая (требование структуризации)

Заключение

- Технология программирования – это
 - а) наука об организации процесса разработки программных продуктов и о применяемых методах и инструментах
 - б) сами методы, инструменты и регламентация процесса разработки
- Хорошая технология нацелена на создание качественного продукта в заданные сроки при заданном бюджете
- Современные ТП используют принципы промышленного производства: стандартизация, автоматизация, специализация исполнителей
- Программные продукты отличаются от других видов промышленной продукции высокой долей интеллектуальной составляющей. Это усложняет проблемы коммуникации и управления проектами
- Долгое время главным показателем качества ПП была эффективность по времени и памяти. Сейчас приоритеты смещены в сторону надежности и удобства пользования

Дополнительные вопросы

1. Что означают аббревиатуры, аналогичные CASE: CAD, CAM, CAE ?
2. В чем отличие понятия *система программирования* от *языка программирования*?
3. Объясните, почему в течение последнего полувека постоянно растут:
 - общий объем ПО в мире, причем быстрее, чем экономика в целом
 - размеры и сложность программ (пример: WinXP vs. DOS)
 - цена программной ошибки (размер ущерба из-за нее)
4. Почему рыночный продукт труднее в производстве, чем заказной?
5. Каковы особые требования к встроенным ПП? Приведите примеры известных вам ПП этого вида.
6. Приведите примеры продуктов для всех четырех комбинаций типов по дихотомийным классификациям 3 (библиотека - приложение) и 4 (прикладной – системный).
7. Что означает "совместимость снизу-вверх" и "сверху-вниз" ?
8. Чем сопрягаемость (interoperability) отличается от стыкуемости (compatibility)?
9. Определите понятие «компьютерная платформа».