

Министерство образования и науки Российской Федерации
КАЗАНСКИЙ ГОСУДАРСТВЕННЫЙ ТЕХНИЧЕСКИЙ УНИВЕРСИТЕТ
им. А.Н. ТУПОЛЕВА

Кафедра прикладной математики и информатики им. Ю.В. Кожевникова

А.И. РАХМАТУЛЛИН

ТЕХНОЛОГИИ РАЗРАБОТКИ ПРОГРАММНЫХ СИСТЕМ

Методические указания для самостоятельной работы студентов

Казань 2008

Содержание

ВВЕДЕНИЕ.....	3
ТЕМА 1. ОСНОВЫ РАЗРАБОТКИ ПО.....	4
ТЕМА 2. МЕТОДОЛОГИЯ РАЗРАБОТКИ ПО.....	6
ТЕМА 3. ТЕХНОЛОГИЯ РАЗРАБОТКИ ПО.....	8
ТЕМА 4. ПОДХОДЫ РАЗРАБОТКИ ПО.....	11
ТЕМА 5. ИНЖЕНЕРИЯ И ИНСТРУМЕНТАРИЙ ПО.....	15
ЛИТЕРАТУРА.....	17

Введение

Самостоятельная работа студентов в процессе изучения дисциплины «Технологии разработки программных систем» подразумевает выполнение следующих работ:

1. Проработка конспекта лекций.
2. Подготовка к лабораторным работам.
3. Выполнение курсовой работы.
4. Изучение дополнительной литературы.

Первая работа выполняется студентом в течение семестра. Содержание второй и третьей работ определяется соответствующими методическими руководствами. Данные методические указания для самостоятельной работы студентов описывают содержание последней из указанных работ.

Тема 1. Основы разработки ПО

Содержание темы

1. Цель и задачи курса.

Структура, цель и задачи курса. Понятие системы и процесса.

Программное обеспечение (ПО): программа, модуль, компонент, пакет, комплекс, средство, документация, система. Программный продукт: изделие, услуга, решение, прототип. Программный проект: команда, заинтересованное лицо, участник проекта, исполнитель, роль, процесс разработки. Жизненный цикл (ЖЦ): ЖЦ проекта, ЖЦ ПО, ЖЦ разработки ПО, ЖЦ системы, ЖЦ разработки системы.

2. Понятие программирования.

Программирование как научная дисциплина. Информатика. Разделы программирования: теория, методология, технология, инженерия, инструментарий программирования. Направления и группы направлений программирования.

Программирование как инженерная деятельность: программирование и разработка ПО. Области разработки ПО. Кризис программирования.

Самостоятельная работа

Основной теоретический материал по данной теме излагается в лекционном курсе. Для более подробного ознакомления с темой необходимо изучить следующую литературу:

1. Учебное пособие [1]: гл. 1, §1.1 «Профессионализм в программировании», §1.5 «Программирование. Искусство, наука или ремесло?».

Подтема «Кризис программирования» раздела 2 целиком вынесена на самостоятельное изучение. Контроль реализуется во время аттестационного тестирования. Необходимый материал представлен в следующей литературе:

1. Учебное пособие [1]: гл. 1, §1.6 «Кризис программирования».
2. Учебник [2]: гл. 1, §1.2 «Проблемы разработки сложных программных систем».

Контрольные вопросы

1. Дайте определение понятиям «алгоритм» и «программа».
2. Дайте определение понятиям «программный модуль» и «программный компонент» как частям программы.
3. Дайте определение понятиям «программная система», «программное обеспечение» (ПО) и «программное средство».
4. Дайте определение понятиям «программный продукт», «услуга», «решение» и «прототип».
5. Дайте определение понятиям «проект» и «команда», «заинтересованное лицо», «участник проекта» и «исполнитель».
6. Дайте определение понятию «роль». Перечислите основные роли участников.
7. В чём заключается концепция жизненного цикла? Дайте определение понятиям, связанным с жизненным циклом проекта.
8. Дайте определение понятиям, связанным с информатикой.
9. Дайте определение понятию «программирование» как научной дисциплине. Перечислите и поясните разделы программирования.
10. Перечислите основные направления программирования и их группы. Сформулируйте рекомендации по их применению при разработке ПО.
11. Дайте определение понятию «программирование» как инженерной деятельности. В чём заключается разница между разработкой ПО и современным пониманием понятия «программирование»?

Тема 2. Методология разработки ПО

Содержание темы

1. Основные понятия и определения.

Методология разработки (методологический подход). Атрибуты методологии. Парадигма программирования. Классификация методологий программирования: ядра и специфики, смешанные и другие методологии. Происхождение методологий: практическая, алгоритмическая и структурно-языковая точки зрения.

2. Методологии программирования.

Ядра методологий: методологии императивного, объектно-ориентированного, функционального, сентенциального, логического и ограничительного программирования. Специфики методологий: методологии структурного (императивного), (императивного) параллельного и логического параллельного программирования.

Самостоятельная работа

Основной теоретический материал по данной теме излагается в лекционном курсе. Для более подробного ознакомления с темой необходимо изучить следующую литературу:

1. Учебное пособие [1]: гл. 2, §2.2 «Ядра методологий», §2.3 «Топологическая специфика методологий», §2.4 «Реализационная специфика методологий» (к разделу 2).

Контрольные вопросы

1. Дайте определение понятию «методология» («методологический подход»).
2. Какие атрибуты связаны с каждой методологией?
3. Охарактеризуйте классификацию методологий.
4. Охарактеризуйте точку зрения на происхождение методологий?
5. В чём суть методологии императивного программирования?
6. В чём суть методологии объектно-ориентированного программирования?
7. В чём суть методологии функционального программирования?

8. В чём суть методологии логического программирования?
9. В чём суть методологии сентенциального программирования?
10. В чём суть методологии ограничительного программирования?
11. В чём суть методологии структурного императивного программирования?
12. В чём суть методологии императивного параллельного программирования?
13. В чём суть методологии логического параллельного программирования?

Тема 3. Технология разработки ПО

Содержание темы

1. Основные понятия и определения.

Жизненный цикл ПО (ЖЦ ПО). Модель ЖЦ.

Технология разработки (технологический подход). Измерения и компоненты технологии разработки: действие, процесс и стадия с их элементами, дисциплина и процедура, методика и практика.

Управление разработкой: ограничения. Формализация разработки: произведённый результат, артефакт и рабочий продукт, базовая линия и базовый план, контрольная точка и веха, итерация и таймбокс.

Классификации процессов и стадий, проектов, подходов и групп подходов.

2. Модели жизненного цикла ПО.

Основные модели ЖЦ. Непланируемая модель ЖЦ. Каскадная модель ЖЦ: классическая (водопад) и модифицированная (водоворот).

Прототипируемая модель ЖЦ, принцип прототипирования. Итеративная инкрементная модель ЖЦ: принципы итеративности и инкрементности. Эволюционная модель ЖЦ, принцип эволюционности.

Спиральная модель ЖЦ: классическая (модель Бозма), модифицированная (современный вариант).

3. Технологические процессы ЖЦ.

Классические процессы ЖЦ: Исследование идеи; Управление; Анализ; Проектирование; Кодирование; Тестирование; Ввод в действие; Сопровождение; Снятие с эксплуатации. Методики анализа и проектирования.

Стандартные процессы ЖЦ. Стандарт ISO/IEC 12207: Архитектура ЖЦ ПО; Основные, вспомогательные и организационные процессы; Адаптация стандарта. Стандарт ISO/IEC 15288: Архитектура ЖЦ системы.

Самостоятельная работа

Основной теоретический материал по данной теме излагается в лекционном курсе. Для более подробного ознакомления с темой необходимо изучить следующую литературу:

1. Обзор [29]: гл. 1, п.1.3.1 «Общие требования к методологии и технологии» (к разделу 1). (Здесь под методологией понимается методика разработки.)

Контрольные вопросы

1. Дайте определение понятиям «жизненный цикл ПО» и «модель ЖЦ».
2. Дайте определение понятию «технология» («технологический подход»).
3. Дайте определение понятиям «процесс» и «стадия».
4. Дайте определение понятиям, связанным со стадией.
5. Дайте определение понятиям «методика» и «практика».
6. Дайте определение понятиям, связанным с формализацией разработки.
7. Перечислите и поясните основные наборы технологических процессов.
8. Приведите классические технологические процессы. Приведите группы стандартных технологических процессов.
9. Перечислите и поясните виды формирования технологических стадий. Приведите классические стадии и фазы.
10. Перечислите классы технологических подходов.
11. Дайте краткое описание основных моделей ЖЦ ПО.
12. Охарактеризуйте классические технологические процессы.
13. Перечислите основные подходы (методики) анализа требований и проектирования для структурной и объектно-ориентированной методологии.
14. Как определяется ЖЦ в стандарте ISO/IEC 12207:1995? Какие элементы ЖЦ выделены в стандарте? Перечислите группы стандартных процессов.
15. Перечислите основные процессы стандарта ISO/IEC 12207:1995.
16. Перечислите вспомогательные процессы стандарта ISO/IEC 12207:1995.
17. Перечислите организационные процессы стандарта ISO/IEC 12207:1995.

18. Перечислите стадии по стандарту ISO/IEC 12207:1995.
19. Приведите группы процессов по стандарту ISO/IEC 15288:2002.
20. Перечислите договорные процессы стандарта ISO/IEC 15288:2002.
21. Перечислите организационные процессы стандарта ISO/IEC 15288:2002.
22. Перечислите проектные процессы стандарта ISO/IEC 15288:2002.
23. Перечислите технические процессы стандарта ISO/IEC 15288:2002.
24. Перечислите специальные процессы стандарта ISO/IEC 15288:2002.
25. Перечислите стадии по стандарту ISO/IEC 15288:2002.

Тема 4. Подходы разработки ПО

Содержание темы

1. Каскадные технологические подходы.

Каскадные подходы: обзорное изложение материала.

2. Каркасные технологические подходы.

Каркасные подходы: общая характеристика подходов.

Унифицированный процесс (UP): обзор подхода. Особенности подхода. Модель ЖЦ (фазы, дисциплины, вехи). Модификации подхода.

Рациональный унифицированный процесс (RUP): обзор подхода, RUP как продукт. Изучение опыта: первопричины и признаки провала проектов. Лучшие практики. Ключевые принципы бизнес-управляемой разработки. Модель ЖЦ (фазы, вехи, дисциплины), итеративность разработки.

Каркас решений Майкрософт (MSF): обзор подхода, MSF как продукт. Основополагающие принципы и ключевые концепции. Модель руководства MSF: особенности подхода. Модель ЖЦ (фазы, вехи, результаты).

Процесс ICONIX (ICONIX Process): обзор подхода, особенности подхода. Суть подхода, ключевые принципы. Модель ЖЦ (этапы, вехи, действия, модели).

3. Эволюционные технологические подходы.

Эволюционные подходы: общая характеристика подходов.

Подходы прототипирования. Эволюционная доставка. Итеративная доставка. Постадийная доставка. Модели ЖЦ для подходов.

Итеративная инкрементная разработка (IID): обзор подхода, циклы PDCA.

Быстрая разработка приложений (RAD): обзор подхода, особенности подхода. Основные принципы. Модель ЖЦ (фазы, деятельности).

4. Адаптивные технологические подходы.

Адаптивные подходы: общая характеристика подходов. Особенности живых подходов: Живой манифест, основные положения и принципы манифеста.

Адаптивная разработка ПО (ASD): обзор подхода, сложные адаптивные системы. Свойства динамического ЖЦ подхода. Модель ЖЦ (фазы, процессы).

Экстремальное программирование (ХР): обзор подхода (проект С3). Категории: ценности, принципы, практики. Модель ЖЦ (фазы, действия, деятельности).

5. Генетические технологические подходы.

Генетические подходы: общая характеристика подходов.

Синтезирующее программирование: задачи синтеза, языки спецификаций.

Конкретизирующее программирование: обобщённое программирование, подходы на основе паттернов и анти-паттернов и на основе архитектурных стилей.

Сборочное программирование: модульное сборочное, объектное сборочное, компонентное сборочное и аспектное сборочное программирование.

6. Формальные технологические подходы.

Формальные подходы: общая характеристика подходов.

Формальные генетические подходы: обзор подходов. Формальное синтезирующее программирование: синтез программы, способы синтеза, творческая составляющая, манипулирование знаниями. Формальное конкретизирующее программирование: конкретизация программы, смешанные вычисления. Формальное сборочное программирование: сборка программы, особенность подхода.

Подходы формальной разработки: формальные методы. Модель ЖЦ (стадии, процессы). Представления системы для языков формальной спецификации. Обзор используемых подходов, подходы исчисления процессов.

Инженерия стерильного цеха (CrSE): обзор подхода (стерильный цех), правила стерильного цеха, особенности подхода. Основные принципы. Модель ЖЦ (фазы, процессы). Методика подхода: стимулы, ответы, последовательности. Метод специфицирования на основе последовательностей. Метод структурирования на основе ящиков.

Самостоятельная работа

Основной теоретический материал по данной теме излагается в лекционном курсе. Для более подробного ознакомления с темой необходимо изучить следующую литературу:

1. Статья [14]: материал о MDA (к разделу 2).

2. Статья [20]: материал о гибких технологических подходах (к разделам 3 и 4).
(Здесь под методологией понимается технология разработки.)
3. Статья [25]: материал о доказательном программировании (к подтеме «Формальные генетические подходы» раздела 6).

Несколько подтем раздела 5 в ряде пунктов целиком вынесено на самостоятельное изучение. Контроль реализуется во время аттестационного тестирования. Необходимый материал представлен в следующей литературе:

1. Учебное пособие [1]: гл. 4, пп.4.4.2.3 – 4.4.2.11 о диаграммах UML (к подтеме «Синтезирующее программирование», п. «Языки спецификаций»).
2. Статья [24]: материал о паттернах проектирования (к подтеме «Конкретизирующее программирование», п. «Подход на основе паттернов»).
3. Учебник [2]: гл. 2, §2.2 «Модули и их свойства» (к подтеме «Сборочное программирование», п. «Модульное сборочное программирование»).
4. Учебное пособие [1]: гл. 4, п.4.3.4 о модульности (к подтеме «Сборочное программирование», п. «Модульное сборочное программирование»).

Контрольные вопросы

1. Охарактеризуйте каскадные технологические подходы. Перечислите виды каскадных подходов и примеры подходов каждого вида.
2. Охарактеризуйте каркасные технологические подходы. Перечислите виды каркасных подходов и примеры подходов каждого вида.
3. Что представляет собой подход УП? Охарактеризуйте подход.
4. Приведите и поясните графическое представление модели ЖЦ для УП.
5. Что представляет собой подход РУП? Охарактеризуйте подход.
6. Приведите и поясните графическое представление модели ЖЦ для РУП.
7. Что представляет собой подход МСФ? Охарактеризуйте подход.
8. Приведите и поясните графическое представление модели ЖЦ для МСФ.
9. Что представляет собой Процесс ICONIX? Охарактеризуйте подход.
10. Приведите и поясните графическое представление модели ЖЦ для ICONIX.

11. Охарактеризуйте эволюционные технологические подходы. Перечислите виды эволюционных подходов и примеры подходов каждого вида.
12. Что представляют собой подходы прототипирования? Перечислите и охарактеризуйте основные подходы прототипирования.
13. Что представляет собой подход ИИР? Охарактеризуйте подход.
14. Что представляет собой подход БРП? Охарактеризуйте подход.
15. Приведите и поясните графическое представление модели ЖЦ для БРП.
16. Охарактеризуйте адаптивные технологические подходы. Перечислите виды адаптивных подходов и примеры подходов каждого вида.
17. Что представляет собой Живая разработка ПО? Опишите Живой манифест.
18. Что представляет собой подход АРП? Охарактеризуйте подход.
19. Приведите и поясните графическое представление модели ЖЦ для АРП.
20. Что представляет собой подход ЭП? Охарактеризуйте подход.
21. Приведите и поясните графическое представление модели ЖЦ для ЭП.
22. Охарактеризуйте генетические технологические подходы.
23. Охарактеризуйте синтезирующее программирование. Опишите языки.
24. Охарактеризуйте конкретизирующее программирование. Опишите подходы.
25. Охарактеризуйте сборочное программирование. Опишите подходы.
26. Охарактеризуйте формальные технологические подходы. Перечислите виды формальных подходов и примеры подходов каждого вида.
27. Охарактеризуйте формальные генетические подходы.
28. Охарактеризуйте подходы формальной разработки. Перечислите языки и подходы формальной разработки.
29. Что представляет собой подход СцИП? Охарактеризуйте подход.
30. Приведите и поясните графическое представление модели ЖЦ для СцИП.

Тема 5. Инженерия и инструментарий ПО

Содержание темы

1. Инженерия ПО.

Стиль программирования: свойства хорошего стиля программирования.

Защитное программирование: основные принципы и механизмы. Проектирование по контракту: используемые виды утверждений.

2. Инструментарий ПО.

Автоматизация разработки. CASE-средства: особенности. Классификация CASE-средств. Системы автоматизации: инструментарий для разработки ПО.

Самостоятельная работа

Основной теоретический материал по данной теме излагается в лекционном курсе. Для более подробного ознакомления с темой необходимо изучить следующую литературу:

1. Учебник [2]: гл. 2, §2.5 «Стиль оформления программы», §2.7 «Программирование "с защитой от ошибок"» (к разделу 1).
2. Учебник [2]: гл. 5, §5.6 «CASE-технологии, основанные на структурных методологиях анализа и проектирования» (к разделу 2).

Подтема «Системы автоматизации» раздела 2 целиком вынесена на самостоятельное изучение. Контроль реализуется во время аттестационного тестирования. Необходимый материал представлен в следующей литературе:

1. Книга [5]: тема «Программная инженерия. Инструменты и методы программной инженерии», §1 «Инструменты программной инженерии».
2. Учебное пособие [1]: гл. 5, §5.2 «Процесс-ориентированный инструментарий» об инструментах для поддержки классических процессов.
3. Учебное пособие [3]: гл. 17 «Автоматизация конструирования визуальной модели программной системы» о средстве Rational Rose.
4. Брошюра [16]: тема «Инструментальные средства IBM Rational».
5. Статья [31]: материал о пакете средств ALM от Borland.

Контрольные вопросы

1. Дайте определение понятию «стиль программирования». Перечислите свойства хорошего стиля программирования.
2. Как формируется стиль программирования? Как связан стиль программирования с методологиями разработки?
3. Дайте определение понятию «защитное программирование». Перечислите основные принципы защитного программирования.
4. Перечислите общие рекомендации по защитному программированию. Перечислите и поясните механизмы защитного программирования.
5. Что представляет собой подход Проектирование по контракту? Поясните механизм, используемый Проектированием по контракту?
6. Что такое CASE-средство? Перечислите их особенности и компоненты.
7. Перечислите основные признаки классификации CASE-средств.
8. Приведите классификацию CASE-средств по типам.
9. Приведите классификацию CASE-средств по категориям.
10. Приведите классификацию CASE-средств по уровням.
11. Кратко охарактеризуйте системы автоматизации.

Литература

Основная литература

1. Одинцов И.О. Профессиональное программирование: Системный подход. СПб.: ВHV-Санкт-Петербург, 2002. 512 с. (Мастер). (Эл. версия, URL: <http://lib.aswl.ru/books/methodology/programming/>). См. также: Одинцов И.О. Профессиональное программирование: Системный подход. 2-е изд., доп. СПб.: ВHV-Санкт-Петербург, 2004. 624 с. (Мастер).
2. Иванова Г.С. Технология программирования: Учеб. для вузов. М.: Изд-во МГТУ им. Н.Э. Баумана, 2002. 320 с. (Информатика в техн. университете).
3. Орлов С.А. Технологии разработки программного обеспечения: Разработка сложных программных систем: Учеб. пособие. 2-е изд. СПб.: Питер, 2003. 480 с.
4. Воройский Ф.С. Информатика: Новый систематизированный толковый словарь-справочник. (Введение в современные информационные и телекоммуникационные технологии в терминах и фактах). 3-е изд., перераб. и доп. М.: ФИЗМАТЛИТ, 2003. 760 с.
5. Орлик С. Введение в программную инженерию и управление жизненным циклом ПО / При уч. Ю. Булуя. (Эл. изд., URL: <http://sorlik.blogspot.com/>).
6. Непейвода Н.Н., Скопин И.Н. Основания программирования. М.-Ижевск: Изд-во РХД, 2003. 880 с. (Эл. версия от 11.09.2003. 2+iv+914 с., URL: <http://ulm.udsu.ru/~nnn/fp.zip>).
7. Соммервилл И. Инженерия программного обеспечения. 6-е изд. / Пер. с англ. под ред. А.А. Минько. М.: Издат. дом «Вильямс», 2002. 624 с.
8. Калянов Г.Н. CASE-технологии: Консалтинг в автоматизации бизнес-процессов. 3-е изд. М.: Горячая линия – Телеком, 2002. 320 с.
9. Кватрани Т. Rational Rose 2000 и UML: Визуальное моделирование / Пер. с англ. М.: ДМК Пресс, 2001. 176 с. (Объектно-ориентированные технологии в программировании).

Дополнительная литература

10. Системный анализ и принятие решений: Словарь-справочник / Под общ. ред. В.Н. Волковой, В.Н. Козлова. М.: Высш. шк., 2004. 616 с.
11. Непейвода Н.Н. Стили и методы программирования // Интернет-университет информационных технологий, 2005. (Проект Изд-ва «Открытые системы»). (Эл. изд., URL: <http://www.intuit.ru/department/se/progstyles/>).
12. Технология разработки программного обеспечения: Конспект лекций / Автор неизвестен. (Эл. изд., URL: <http://www.solomil.ru/>).
13. Берлинский К. Набор серебряных пуль: Справочник удачных проектных решений при разработке ПО. 2004. 99 с. (Эл. изд., v.1.37, 20.06.2004).
14. Марков Е. Архитектура, управляемая моделью // CIT City. 2005. (Эл. изд.: 15.12.2005, URL: <http://citcity.ru/>).
15. Безуглый Д.Л. Технология разработки программного обеспечения // Корпоративные Информационные Системы. 2001. №2. С. 24–30.
16. IBM Corp. Методология и инструментальные средства IBM Rational для разработки программных систем. 26.04.2007 / Пер. с англ. (Эл. изд., URL: <http://1050049.ru/iservices/files.asp?artId=2186&file=0>).
17. Microsoft Corp. Microsoft Solutions Framework: Белая книга. Модель процессов MSF. Ver. 3.1, 06.2002 / Пер. с англ. под ред. В. Павлова. (Эл. изд., URL: http://www.microsoft.com/rus/docs/msdn/msf/MSF_process_model_rus.doc).
18. Коуберн А. Четвёртое измерение, или Как обмануть Железный Треугольник / Пер. с англ. К. Максимова, А. Максимова. 08.08.2004. (Эл. изд., оригинал от 04.10.2003, URL: <http://www.maxkir.com/>).
19. Коуберн А. Создание программного обеспечения как коллективная игра (По материалам статей А. Коуберна за 1997 – 2004 гг.) / Пер. с англ. К. Максимова, А. Максимова. 11.09.2004. (Эл. изд., URL: <http://www.maxkir.com/>).
20. Фаулер М. Новые методологии программирования / Пер. с англ. К. Максимова, А. Максимова. 13.10.2001. (Эл. изд., URL: <http://www.maxkir.com/>).

21. Хайсмит Дж. Устаревшие методологии – на пенсию! / Пер. с англ. К. Максимова, А. Максимова. 20.03.2002. (Эл. изд., оригинал от 07 – 08.2000, URL: <http://www.maxkir.com/>).
22. Бек К. Экстремальное программирование. СПб.: Питер. 216 с. (Б-ка программиста). См. также: Бек К. Экстремальное программирование: Разработка через тестирование. СПб.: Питер, 2003. 224 с. (Б-ка программиста).
23. Гамма Э., Хелм Р., Джонсон Р., Влиссидес Дж. Приёмы объектно-ориентированного программирования: Паттерны проектирования / Пер. с англ. А. Слинкина. СПб.: Питер, 2001. 368 с. (Б-ка программиста).
24. Дубина О. Обзор паттернов проектирования // CIT Forum. 2005. (Эл. изд., URL: <http://citforum.ru/SE/project/pattern/>).
25. Ершов А.П. Научные основы доказательного программирования. 1984. 15 с. (Эл. версия, URL: <http://www.europrog.ru/paper/ae1984-02r.pdf>).
26. Ершов А.П. Отношения методологии и технологии программирования // Технология программирования. Информ. материалы и тез. плен. докл. II Всесоюз. конф. (Киев, 18 – 21.11.1986.) / Ин-т кибернетики им. В.М. Глушкова, АН УССР. Киев, 1986. С. 10 – 13. (Эл. версия, URL: <http://erшов.iis.nsk.su/>).
27. Дейкстра Э.В. Дисциплина программирования / Пер. с англ. под ред. Э.З. Любимского. М.: Мир, 1978. 290 с. (Математическое обеспечение ЭВМ).
28. Грис Д. Наука программирования / Пер. с англ. под ред. А.П. Ершова. М.: Мир, 1984. 416 с.
29. Вендров А.М. CASE-технологии: Современные методы и средства проектирования информационных систем // CIT Forum. 1997. (Эл. изд., URL: <http://citforum.ru/>).
30. Норенков И.П. Основы автоматизированного проектирования: Учеб. для вузов. 2-е изд., перераб. и доп. М.: Изд-во МГТУ им. Н.Э. Баумана, 2002. 336 с. (Информатика в техн. университете).
31. Дубова Н. В круге разработки // Открытые системы. 2003. № 9; CIT City. (Эл. изд.: 18.09.2003, URL: <http://www.osp.ru/os/>).

Документация

Основные стандарты, связанные с жизненным циклом ПО и систем:

32. ISO/IEC 12207:1995 «Information Technology – Software Life Cycle Processes». Рус. версия: ГОСТ Р ИСО/МЭК 12207:1999 «Информационная технология. Процессы жизненного цикла программных средств».
33. ISO/IEC 15288:2002 «Systems Engineering – System Life Cycle Processes». Рус. версия: ГОСТ Р ИСО/МЭК 15288:2005 «Системная инженерия. Процессы жизненного цикла систем».
34. ISO/IEC TR 15271:1998 «Information Technology – Guide for ISO/IEC 12207 (Software Life Cycle Processes)». Рус. версия: ГОСТ Р ИСО/МЭК ТО 15271:2002 «Информационная технология. Руководство по применению ГОСТ Р ИСО/МЭК 12207 (Процессы жизненного цикла программных средств)».
35. ISO/IEC TR 16326:1999 «Software Engineering – Guide for the Application of ISO/IEC 12207 to Project Management». Рус. версия: ГОСТ Р ИСО/МЭК ТО 16326:2002 «Программная инженерия. Руководство по применению ГОСТ Р ИСО/МЭК 12207 при управлении проектом».
36. ISO/IEC TR 19760:2003 «Systems Engineering – A Guide for the Application of ISO/IEC 15288 (System Life Cycle Processes)». Рус. версия: ГОСТ Р ИСО/МЭК ТО 19760:2003 «Системная инженерия. Руководство по применению ГОСТ Р ИСО/МЭК 15288 (Процессы жизненного цикла систем)».

Другая документация по разработке ПО и систем:

37. IEEE Guide to the Software Engineering Body of Knowledge (SWEBOK). 2004 Version. IEEE Computer Society. Los Alamos, CA. 2004. 204 p.

Интернет – источники

38. Объектно-ориентированный анализ и проектирование. (обозначение сайта – ООА&П / OOA&D, URL: <http://ood.asf.ru/>).
39. Клуб разработчиков программных систем. (URL: <http://www.caseclub.ru/>).

40. Сайт MAXKIR.com – переводы зарубежных публикаций К. Максимова, А. Максимова. (URL: <http://www.maxkir.com/>)
41. Wikipedia.org – The Free Encyclopedia. (URL: <http://en.wikipedia.org/>).
42. Европейский центр программирования. (URL: <http://www.europrog.ru/>).
43. Agile Manifesto – Живой манифест. (URL: <http://www.agilemanifesto.org/>).
44. PraxOS – Организационная система PraxOS. (URL: <http://praxos.ru/>).
45. ИНТУИТ – Интернет-университет информационных технологий. (URL: <http://www.intuit.ru/>).
46. Open Systems – Проект «Открытые системы». (URL: <http://www.osp.ru/>).
47. CIT – Центр информационных технологий. (URL: <http://www.citforum.ru/>, <http://www.citcity.ru/> и другие сайты).

