

Основы программной инженерии (ПОИТ)

1. Дисциплина: «Основы программной инженерии» для специальности ПОИТ
Семестр I.

Всего 108 часов, из них лекций 36 часов, лабораторных 36 часов, *экзамен*.

Лектор: *Наркевич Аделина Сергеевна*, ст. преподаватель, кафедры программной инженерии (а.408, к.1).

email: narkevich.adelina@gmail.com

Цели и задачи курса.

Целью курса является ознакомление с принципами организации и создания надежного, качественного программного обеспечения, удовлетворяющего предъявляемым к нему требованиям.

В рамках изучения курса предполагается решение следующих задач:

- рассмотрение технологических основ процесса разработки программного обеспечения;
- изучение основ унифицированного языка UML для визуального моделирования элементов предметной области в рамках проектирования программной системы и ее основных компонент;
- получение практического опыта работы в команде из 5 человек;
- приобретение навыков анализа, проектирования, документирования и разработки небольших программных комплексов.

Лекции и задания для лабораторных работ доступны в электронном виде:

<https://diskstation.belstu.by:5001/>

логин: student

пароль fitfit

папка:

/Для_студентов_ФИТ_БГТУ/ПРЕПОДАВАТЕЛИ/Наркевич/
ОПИ, 1 курс (ПОИТ)

2. Литература:

Основная литература:

1. Орлов, С. А. Программная инженерия / С. А. Орлов. – Санкт-Петербург : Питер, 2016. – 640 с.
2. Чакон С., Штрауб Б. Git для профессионального программиста. – СПб.: Питер, 2016. — 496 с.
3. Лаврищева, Е.М. Программная инженерия. Парадигмы, технологии и CASE-средства: учебник для вузов / Е.М.Лаврищева. –2-е изд., испр. и доп. – М.: Издательство Юрайт, 2016. – 280 с. – Серия: Университеты России.
4. Липаев, В. В. Программная инженерия. Методологические основы / В. В. Липаев. – М. : ТЕИС, 2006. – 608 с.
5. Липаев, В. В. Процессы и стандарты жизненного цикла сложных программных средств: справочник / В. В. Липаев. – М. : Синтег, 2006. – 276 с.
6. Мацяшек, Л. А. Практическая программная инженерия на основе учебного примера / Л. А. Мацяшек. – М. : БИНОМ, 2009. – 956 с.
7. Вигерс, К. И. Разработка требований к программному обеспечению / К. И. Вигерс. – М. : Русская редакция, 2004. – 576 с.

Дополнительная литература:

8. Батоврин, В. К. Толковый словарь по системной и программной инженерии / В. К. Батоврин. – М. : ДМК Пресс, 2012. – 280 с.
9. Единая система программной документации. ИПК Издательство стандартов, 2001. – 164 с.
10. Городняя, Л. В. Парадигма программирования : курс лекций / Л. В. Городняя ; Новосиб. гос. ун-т. – Новосибирск : РИЦ НГУ, 2015. – 206 с.
11. Макконнел С. Профессиональная разработка программного обеспечения. – СПб., Питер, 2007 – 240 с.

Электронные ресурсы

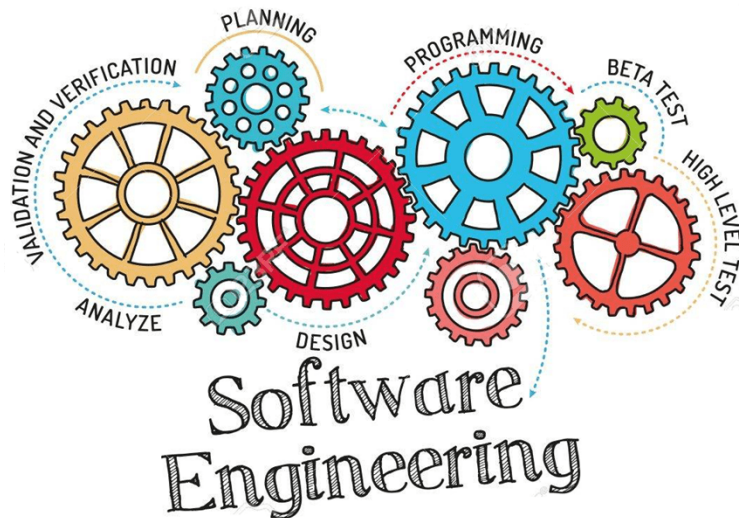
12. Git How To [Электронный ресурс]. – Режим доступа: <https://githowto.com/ru/> – Дата доступа: 24.06.2021.
13. Pro Git [Электронный ресурс]. – Режим доступа: <https://git-scm.com/book/ru/v2/> – Дата доступа: 24.06.2021.

Основы программной инженерии (ПОИТ)

ВВЕДЕНИЕ. ОБЩИЕ СВЕДЕНИЯ О ПРОГРАММНОЙ ИНЖЕНЕРИИ (SOFTWARE ENGINEERING)

План лекции:

- становление программной инженерии;
- основные определения;
- отличия от других инженерий;
- профессиональные и этические требования.



В эпоху информационного общества (общества, основанного на знаниях):

производство программного обеспечения (ПО) является крупнейшей отраслью мировой экономики

Количество программистов:

Общая статистика: Ожидается, что к концу 2025 года глобальное население программистов достигнет 45 миллионов человек.

США: ИТ-сообщество насчитывает 7,7 млн человек, связанных с ИТ-сферой.

Россия: число ИТ-специалистов в 2025 году выросло на 14% и составило более 1 млн человек.

Беларусь: число ИТ-специалистов в секторе «информация и связь» составило 74,3 тыс. человек (данные Белстата на июль 2025).

Программная инженерия (промышленное программирование) ассоциируется с разработкой сложных программ коллективами разработчиков.

Проблемы становления и развития отрасли – высокая стоимость программного обеспечения, сложность его создания, необходимость управления и прогнозирования процессов разработки.

Цель программной инженерии – сокращение стоимости программ.

История

Понятие «информационное общество» зародилось в 1940-х гг. с появлением кибернетики и связано с именами ученых:

 <p>Клод Шенон - создатель теории информации, 1948 г.</p>	 <p>Норберт Винер - основоположник кибернетики и теории искусственного интеллекта</p>	 <p>Джон фон Нейман - создатель современной архитектуры компьютеров</p>	 <p>Алан Тьюринг предложил в 1936 году абстрактную вычислительную «Машину Тьюринга» - модель компьютера общего назначения, которая позволила формализовать понятие алгоритма</p>	 <p>Андрей Николаевич Колмогоров - один из самых выдающихся математиков XX века. Им получены фундаментальные результаты в математической логике и др., внес важный вклад в теорию информации, теорию сложности алгоритмов.</p>
---	--	--	--	---

Программирование – стадии эволюции

50-е годы 20 века.

Программирование в машинном коде для решения, главным образом, научно-технических задач (расчет по формулам).

Наличие достаточно четко сформулированного технического задания.

Отсутствие этапа проектирования.

Составление документации после завершения разработки.

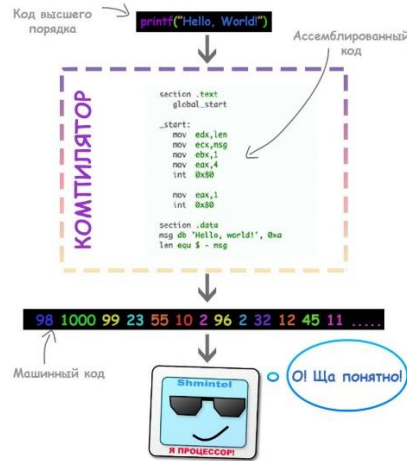
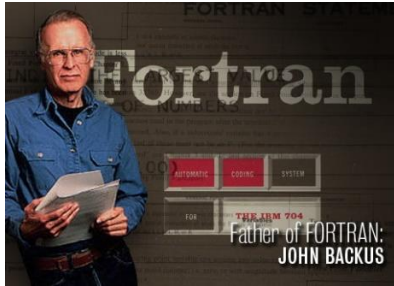
Зарождение концепции модульного программирования.

<div><div>ПРОЦЕССОР</div><div><div>Исполнитель управления</div><div>Рег. В</div><div>Рег. С</div><div>Рег. А</div><div>Рез. счет. "0" → 1</div><div>Σ</div><div>Ариф. лог. устр.</div></div></div> <div><div>ПАМЯТЬ</div><div><div>ПАМЯТЬ КОМАНД</div><table><thead><tr><th>Адр.</th><th>Код</th><th>Адрес</th></tr></thead><tbody><tr><td>1</td><td>01</td><td>01</td></tr><tr><td>2</td><td>01</td><td>01</td></tr><tr><td>3</td><td>02</td><td>В. С. А.</td></tr><tr><td>4</td><td>01</td><td>А. 01</td></tr><tr><td>5</td><td>01</td><td>01</td></tr><tr><td>6</td><td>03</td><td>В. А.</td></tr><tr><td>7</td><td>01</td><td>А. 02</td></tr><tr><td>8</td><td>04</td><td>2</td></tr><tr><td>9</td><td>PS</td><td>STOP</td></tr></tbody></table></div><div><div>ПАМЯТЬ ДАННЫХ</div><table><thead><tr><th>Адр.</th><th>Код</th><th>Адрес</th></tr></thead><tbody><tr><td>01</td><td>01</td><td>01</td></tr><tr><td>02</td><td>01</td><td>01</td></tr><tr><td>03</td><td>02</td><td>В. С. А.</td></tr><tr><td>04</td><td>01</td><td>А. 01</td></tr><tr><td>05</td><td>01</td><td>01</td></tr><tr><td>06</td><td>03</td><td>В. А.</td></tr><tr><td>07</td><td>01</td><td>А. 02</td></tr><tr><td>08</td><td>04</td><td>2</td></tr><tr><td>09</td><td>PS</td><td>STOP</td></tr></tbody></table></div></div> <div><div>Толк. операций:</div><div>01 Переписать данные</div><div>02 Считать содержимое регистра</div><div>03 Прибавить (-1) к содержимому регистра</div><div>04 Начать выполнение команд с указанного адреса, если содержимое рег. состояния = 1</div><div>05 Останов</div></div>	Адр.	Код	Адрес	1	01	01	2	01	01	3	02	В. С. А.	4	01	А. 01	5	01	01	6	03	В. А.	7	01	А. 02	8	04	2	9	PS	STOP	Адр.	Код	Адрес	01	01	01	02	01	01	03	02	В. С. А.	04	01	А. 01	05	01	01	06	03	В. А.	07	01	А. 02	08	04	2	09	PS	STOP	<table><thead><tr><th>Машинный код</th><th>Ассемблер</th></tr></thead><tbody><tr><td>0E</td><td>PUSH CS</td></tr><tr><td>1F</td><td>POP DS</td></tr><tr><td>BA0E00</td><td>MOV DX, 000E</td></tr><tr><td>B409</td><td>MOV AH, 09</td></tr><tr><td>CD21</td><td>INT 21</td></tr><tr><td>B8014C</td><td>MOV AX, 4C01</td></tr><tr><td>CD21</td><td>INT 21</td></tr><tr><td>54</td><td>PUSH SP</td></tr><tr><td>68</td><td>DB 68</td></tr><tr><td>69</td><td>DB 69</td></tr><tr><td>7320</td><td>JNB 0033</td></tr><tr><td>7072</td><td>JO 0087</td></tr><tr><td>6F</td><td>DB 6F</td></tr><tr><td>67</td><td>DB 67</td></tr><tr><td>7261</td><td>JB 007A</td></tr><tr><td>6D</td><td>DB 6D</td></tr><tr><td>206361</td><td>AND [BP+DI+61], AH</td></tr><tr><td>6E</td><td>DB 6E</td></tr><tr><td>6E</td><td>DB 6E</td></tr><tr><td>6F</td><td>DB 6F</td></tr></tbody></table>	Машинный код	Ассемблер	0E	PUSH CS	1F	POP DS	BA0E00	MOV DX, 000E	B409	MOV AH, 09	CD21	INT 21	B8014C	MOV AX, 4C01	CD21	INT 21	54	PUSH SP	68	DB 68	69	DB 69	7320	JNB 0033	7072	JO 0087	6F	DB 6F	67	DB 67	7261	JB 007A	6D	DB 6D	206361	AND [BP+DI+61], AH	6E	DB 6E	6E	DB 6E	6F	DB 6F	<div><div>Модули</div><div><div>Основная программа</div><div>Данные</div></div><div><div>Данные</div><div>Данные</div><div>Данные</div><div>Данные</div><div>Данные</div></div><div><div>Подпрограммы с локальными данными</div></div></div>
Адр.	Код	Адрес																																																																																																						
1	01	01																																																																																																						
2	01	01																																																																																																						
3	02	В. С. А.																																																																																																						
4	01	А. 01																																																																																																						
5	01	01																																																																																																						
6	03	В. А.																																																																																																						
7	01	А. 02																																																																																																						
8	04	2																																																																																																						
9	PS	STOP																																																																																																						
Адр.	Код	Адрес																																																																																																						
01	01	01																																																																																																						
02	01	01																																																																																																						
03	02	В. С. А.																																																																																																						
04	01	А. 01																																																																																																						
05	01	01																																																																																																						
06	03	В. А.																																																																																																						
07	01	А. 02																																																																																																						
08	04	2																																																																																																						
09	PS	STOP																																																																																																						
Машинный код	Ассемблер																																																																																																							
0E	PUSH CS																																																																																																							
1F	POP DS																																																																																																							
BA0E00	MOV DX, 000E																																																																																																							
B409	MOV AH, 09																																																																																																							
CD21	INT 21																																																																																																							
B8014C	MOV AX, 4C01																																																																																																							
CD21	INT 21																																																																																																							
54	PUSH SP																																																																																																							
68	DB 68																																																																																																							
69	DB 69																																																																																																							
7320	JNB 0033																																																																																																							
7072	JO 0087																																																																																																							
6F	DB 6F																																																																																																							
67	DB 67																																																																																																							
7261	JB 007A																																																																																																							
6D	DB 6D																																																																																																							
206361	AND [BP+DI+61], AH																																																																																																							
6E	DB 6E																																																																																																							
6E	DB 6E																																																																																																							
6F	DB 6F																																																																																																							

60-е годы.

Широкое использование языков программирования высокого уровня (Алгол 60, Фортран, Кобол и др.).

Возрастание сложности задач, решаемых с помощью компьютеров. Использование методов коллективной работы при создании больших программных систем.



70-е годы.

Широкое распространение информационных систем и баз данных.

Развитие абстрактных типов данных.

Исследование проблем обеспечения надежности и мобильности программных средств.

Создание методики управления коллективной разработкой программ.

Появление инструментальных средств поддержки программирования.

БД

содержит обширную информацию самого различного типа: текстовую, графическую, звуковую, мультимедийную.

Примеры:

1. в БД законов – тексты самих законов,
2. в БД эстрадной песни – тексты и ноты песен, биографию авторов, информация о поэтах, композиторах и исполнителях, звуковые и видеоклипы.

Абстрактный тип данных – это совокупность данных и операций над ними

Структура данных является частью реализации АДТ
Перед реализацией АДТ необходимо тщательно описать все операции, которые необходимо выполнять



80-е годы.

Широкое внедрение персональных компьютеров во все сферы человеческой деятельности.

Бурное развитие пользовательских интерфейсов и создание четкой концепции качества ПО.

Внедрение объектного подхода к разработке программных систем.

Развитие концепции компьютерных сетей.



Интерфейс включает в себя:

- способы взаимодействия с внутренней частью программы (операционной системой, платформой, сервером и т.д.);
- дизайн;
- доступные функции.

Топология сети – схема соединения компьютеров в сети



90-е годы.

В 1989 году реализован проект Всемирной паутины.

Актуальность проблемы защиты компьютерной информации и передаваемых по сети сообщений.

Развитие CASE-средств разработки программного обеспечения.

Глобальная сеть Internet

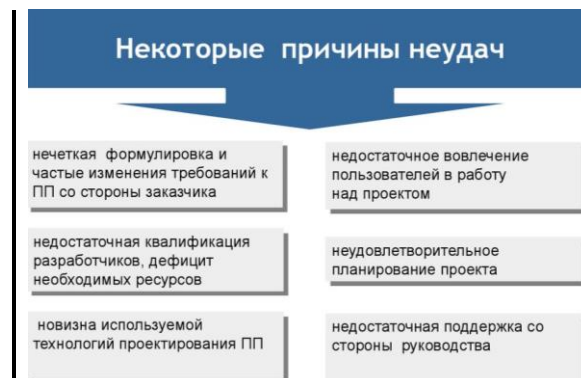


Разработка ПО по-прежнему остается непредсказуемой

Процент успешных проектов по созданию программного обеспечения достаточно низок



Некоторые причины неудач



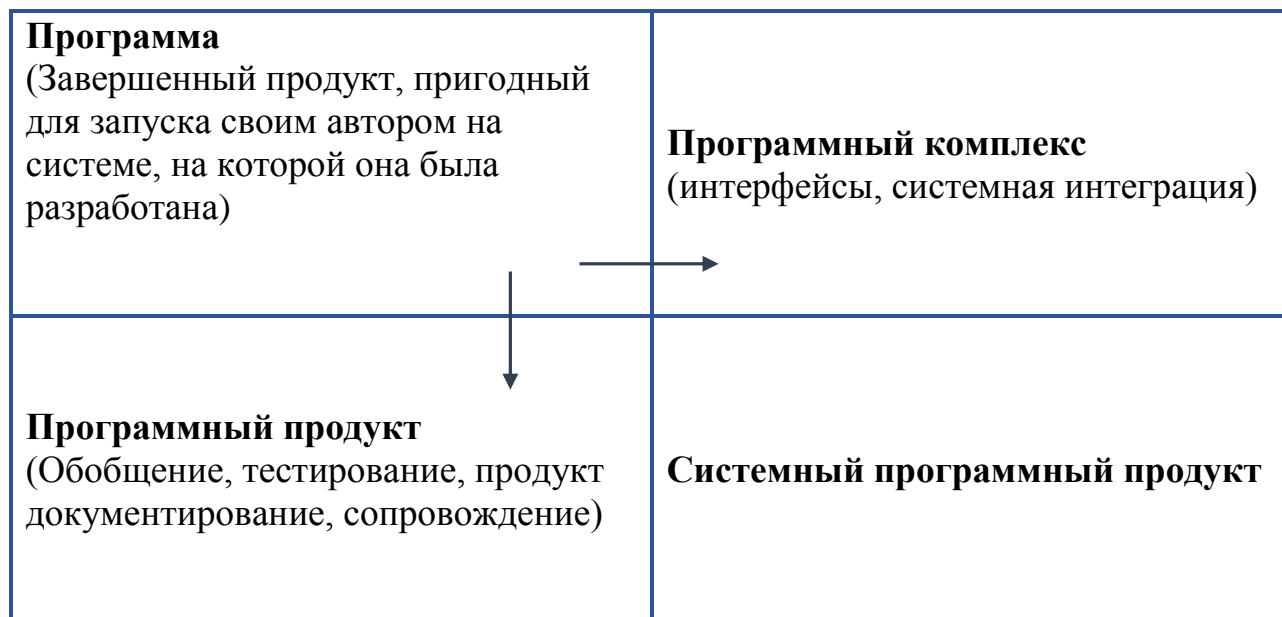
Предпосылки

Повторное использование кода (модульное программирование)

Рост сложности программ (структурное программирование)

Модификация программ (ООП)

Цикл разработки программной системы:



Основные определения:

Программа – это объект разработки, который не является осязаемым (нельзя пощупать, взвесить и т. п.), доступен пониманию ЭВМ, для которой написан.

Свойства хорошей программы

- Выполнение функциональных требований
- Соответствие нефункциональным требованиям
- Сопровождаемость (maintainability)
- Надежность (dependability)
- Эффективность (efficiency)
- Удобство использования (usability)

Программный продукт (ПП): программа, работающая без авторского присутствия. Программный продукт исполняется, тестируется, конфигурируется без присутствия автора и сопровождается документацией.

Программное обеспечение (ПО) – совокупность программ системы обработки информации и программных документов, необходимых для эксплуатации этих программ (ГОСТ 19781-90)

Программная инженерия (Software Engineering) ориентирована на разработку программного обеспечения прикладных и информационных систем разного назначения.

Определение из свода знаний по программной инженерии SWEBOOK:

- 1) **Программная инженерия** – это применение систематического, дисциплинированного и измеряемого подхода к разработке, эксплуатации и сопровождению программного обеспечения (ПО) с применением инженерных методов к разработке ПО.
- 2) **Программная инженерия** – учебная дисциплина, изучающая указанные выше подходы.

Отличие от других инженерий

Программная инженерия – это система методов, средств и дисциплин планирования, разработки, эксплуатации и сопровождения программного обеспечения, готового к внедрению.

Программа – не материальный объект: фазы производства и изготовления образца отсутствуют.

Стоимость программы зависит от стоимости и качества проектирования.

Нет объективных законов контроля проекта: тестирование – единственный способ проверки

Теоретический фундамент программной инженерии

Главные положения фундаментальных наук: теория алгоритмов, математическая логика теория управления теории множеств, и т.п.

Формальные методы программирования: спецификация программ, их доказательство, верификация и тестирование, математические модели надежности, риска и т.п.;

Прикладные методы: приемы, принципы, правила, отдельные действия и цельные процессы жизненного цикла (ЖЦ) производства компьютерных систем, которые являются инструментами коллективной разработки, применяемыми исполнителями крупных программных проектов;

Методы управления коллективами: планирование по сетевым графикам, контроль работ в процессах ЖЦ, измерение и оценка качества промежуточных результатов производства, прогнозирования и регулирования сроков и стоимости изготовления продукта, а также его сертификации.

Успех реализации проекта ПО обусловлен пятью взаимосвязанными аспектами:



Жизненный цикл ПО – непрерывный процесс с момента принятия решения о создании ПО до снятия его с эксплуатации.

Программная инженерия (Software Engineering)



Стандарты программной инженерии

Стандарт (standard) – норма, образец, мерило

- нормативно-технический документ, устанавливающий нормы и правила по отношению к объекту стандартизации, утверждается компетентным органом;
- типовой образец, эталон, модель, принимаемые за исходные для сопоставления с ними других объектов.

Основные типы стандартов

Корпоративные стандарты разрабатываются крупными фирмами с целью повышения качества своей продукции. Создаются на основе собственного опыта компании, но с учетом требований мировых стандартов. Не сертифицируются, но являются обязательными для применения внутри корпорации.

Отраслевые стандарты действуют в пределах организаций некоторой отрасли (министерства). Разрабатываются с учетом требований мирового опыта и специфики отрасли. Являются обязательными для отрасли. Подлежат сертификации.

Государственные стандарты (ГОСТы) принимаются государственными органами и имеют силу закона. Разрабатываются с учетом мирового опыта или на основе отраслевых стандартов. Могут иметь как рекомендательный, так и обязательный характер. Для сертификации создаются государственные или лицензированные органы сертификации.

Международные стандарты разрабатываются специальными международными организациями на основе мирового опыта и лучших корпоративных стандартов. Имеют сугубо рекомендательный характер.

Разработчики стандартов в области программной инженерии

ISO – The International Standards Organization международная организация по стандартизации, работающая в сотрудничестве с **IEC – The International Electrotechnical Commission** – международной электротехнической комиссией

IEEE Computer Society – профессиональное объединение специалистов в области программной инженерии

ACM – Association for Computing Machinery – Ассоциация по вычислительной технике

SEI – Software Engineering Institute – Институт Программной Инженерии при университете Карнеги Мелон

PMI – Project Management Institute – Международный Институт Проектного Менеджмента

Объекты стандартизации в программной инженерии

- процессы разработки ПО;
- продукты разработки;
- ресурсы, которые используют процессы для создания программного продукта.

Основные стандарты программной инженерии

ISO/IEC 12207 – Information Technology – Software Life Cycle Processes – процессы жизненного цикла программных средств.

SEI CMM – Capability Maturity Model (for Software) – модель зрелости процессов разработки программного обеспечения.

ISO/IEC 15504 – Software Process Assessment – оценка и аттестация зрелости процессов создания и сопровождения ПО. Является развитием и уточнением ISO 12207 и SEI CMM.

PMBOK – Project Management Body of Knowledge – свод знаний по управлению проектами.

SWEBOK – Software Engineering Body of Knowledge – свод знаний по программной инженерии.

ACM/IEEE CC2001 – Computing Curricula 2001 – академический образовательный стандарт в области компьютерных наук.

Ядро профессиональных знаний SWEBOK (Software Engineering Body of Knowledge)

Software Requirements – требования к ПО

Software Design – проектирование ПО

Software Construction – конструирование ПО

Software Testing – тестирование ПО

Software Maintenance – сопровождение ПО

Software Configuration Management – управление конфигурацией

Software Engineering Management – управление ИТ проектом

Software Engineering Process – процесс программной инженерии

Software Engineering Tools and Methods – методы и инструменты

Software Quality – качество ПО

Свод знаний по управлению проектами PMI PMBOK (Project Management Body of Knowledge)

Управление интеграцией – Project Integration Management

Управление содержанием – Project Scope Management

Управление временем – Project Time Management

Управление затратами – Project Cost Management

Управление рисками – Project Risk Management

Управление персоналом – Project Personnel Management

Управление коммуникациями – Project Communication Management

Управление закупками – Project Procurement Management

Управление качеством – Project Quality Management

Кодекс этики программной инженерии (краткая версия)

- **программные инженеры** будут действовать соответственно общественным интересам;
- **программные инженеры** будут действовать в интересах клиентов и работодателя, соответственно общественным интересам;
- **программные инженеры** будут добиваться, чтобы произведенные ими продукты и их модификации соответствовали высочайшим профессиональным стандартам;
- **программные инженеры** будут добиваться честности и независимости в своих профессиональных суждениях;
- **менеджеры и лидеры программных инженеров** будут руководствоваться этическим подходом к руководству разработкой и сопровождением ПО, а также будут продвигать и развивать этот подход;
- **программные инженеры** будут улучшать целостность и репутацию своей профессии соответственно с интересами общества;
- **программные инженеры** будут честными по отношению к своим коллегам и будут всячески их поддерживать;
- **программные инженеры** в течение всей своей жизни будут учиться практике своей профессии и будут продвигать этический подход к практике своей профессии.

Принципы, положенные в основу кодекса этики программной инженерии

- согласование профессиональной деятельности инженеров-программистов с интересами общества;
- взаимоотношения между клиентом, работодателем и исполнителем разработки;
- достижение соответствия качества продукта лучшим профессиональным стандартам;
- честность и независимость профессиональных оценок;
- соблюдение этических норм в менеджменте и в
- сопровождении разработок;
- поддержка становления профессии в соответствии с кодексом этики;
- соблюдение этических норм во взаимоотношениях с коллегами;
- усовершенствование специальности