

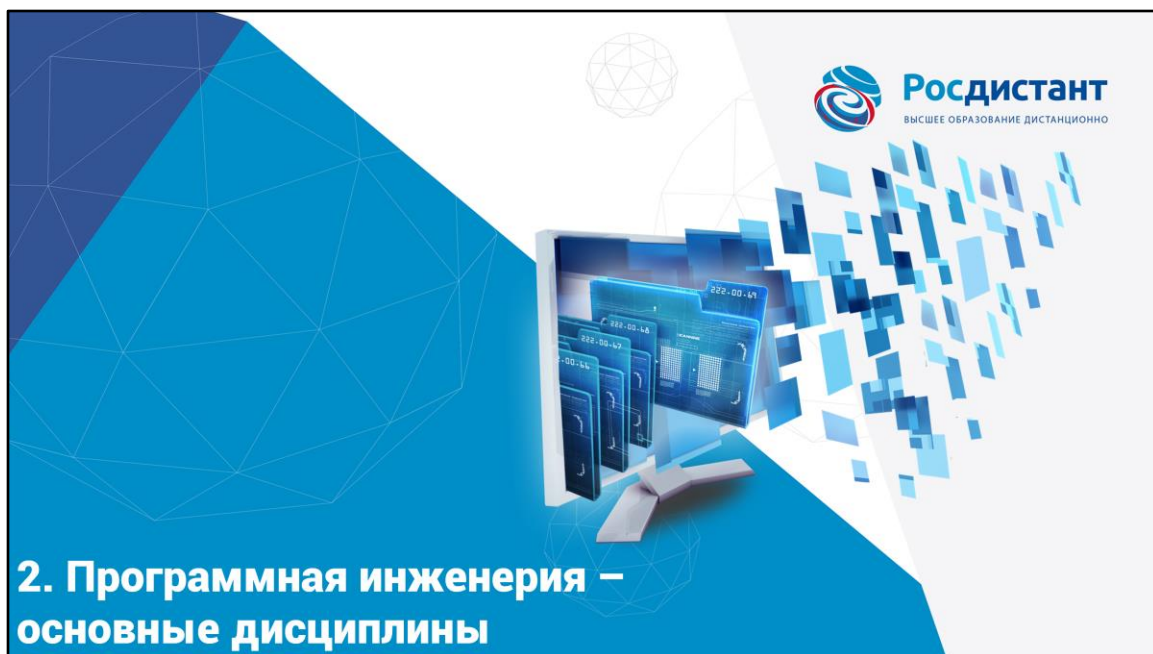


Росдистант
ВЫСШЕЕ ОБРАЗОВАНИЕ ДИСТАНЦИОННО

ВВЕДЕНИЕ В ПРОГРАММНУЮ

ИНЖЕНЕРИЮ 2 ЧАСТЬ



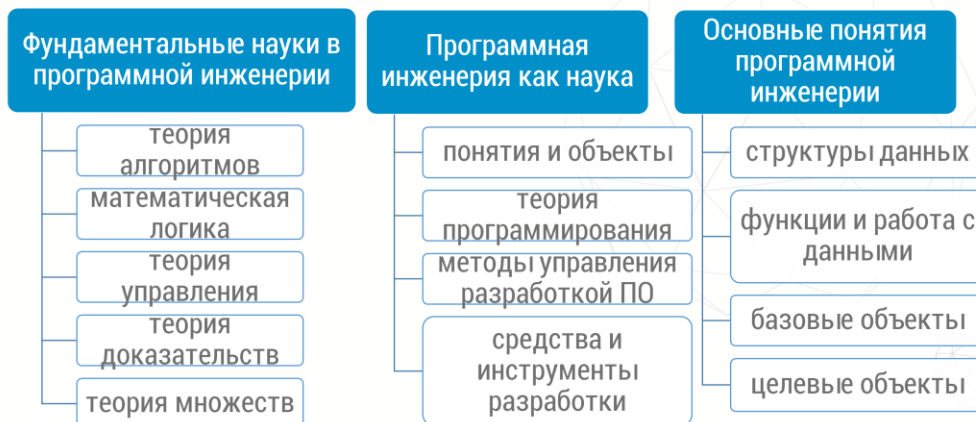


2. Программная инженерия – основные дисциплины

Тема 2. Программная инженерия: содержание дисциплины

В ходе изучения данной темы у вас должно сложиться общее представление о программной инженерии как науке. Вы должны освоить основные стандарты разработки программного обеспечения, ознакомиться с принципами программной инженерии, а также кодексом этики и профессиональной практики разработки программного обеспечения.

ПРОГРАММНАЯ ИНЖЕНЕРИЯ



Цель программной инженерии как науки: применение знаний, полученных из фундаментальных наук, для построения сложных программных продуктов. Построение программного продукта включает анализ автоматизируемой предметной области и производство кода для выполнения на компьютере. Фундаментальные науки, используемые в программной инженерии, следующие.

- Теория алгоритмов – нормальные алгоритмы, вычислимые функции, машина Тьюринга, граф-схемы, модели алгоритмов.
- Математическая логика – формальный вывод утверждений.
- Теория управления – принципы, методы и общие законы планирования и управления в сложных системах.
- Теория доказательств – математическая теория вывода по аксиомам и утверждениям, теория верификации программ.
- Теория множеств – формальное представление совокупностей объектов из предметной области.

Программная инженерия как наука содержит в себе:

- основные понятия и объекты;
- теорию программирования;
- методы управления изготовлением программного обеспечения;

- средства и инструменты процессов изготовления.

К основным понятиям программной инженерии относятся:

- данные и структуры данных;
- функции и композиции над данными;
- базовые объекты – модули, компоненты, каркасы, контейнеры и тому подобное;
- целевые объекты – программное обеспечение, программная система, семейство систем, программный проект и тому подобное.

Целевые объекты изготавливаются из базовых при помощи инженерных методов, включая управление сроками и затратами на производство.

Таким образом, мы видим, что программная инженерия – многогранная наука, которая сопряжена с другими дисциплинами и базируется на ключевых понятиях, являющихся базовыми для всех информационных наук.

ЦЕЛЕВЫЕ ОБЪЕКТЫ ПРОГРАММНОЙ ИНЖЕНЕРИИ

Программная система - комплекс взаимосвязанных программ и средств, реализующих набор функций предметной области

Программное обеспечение - совокупность программных инструментов, реализующих определенный набор функций в компьютерной системы

Семейство систем - совокупность программных систем с характеристиками, которые удовлетворяют заданным потребностям предметной области

Теория программирования - совокупность методов, языков и средств разработки программного обеспечения как объекта программной инженерии



К целевым объектам программной инженерии относятся следующие.

- Программная прикладная система – комплекс интегрированных программ и средств, реализующих набор взаимосвязанных функций некоторой предметной области в заданной среде.
- Программное обеспечение – совокупность программных инструментов, реализующих определенную функцию компьютерной системы.
- Семейство систем – совокупность программных систем с общими и переменными характеристиками, удовлетворяющими заданные потребности предметной области.
- Теория программирования – совокупность методов, языков и средств описания и проектирования целевых объектов программной инженерии, а также методы их доказательства, верификации и тестирования.

Составляющие теории программирования:

- методы программирования – теоретические и прикладные;
- методы проверки правильности;
- методы оценки промежуточных результатов проектирования и конечного продукта относительно показателей: надежность, качество, точность, продуктивность и так далее;
- методы управления и контроля разработки.

Но это отнюдь не полный список целевых объектов. В процессе разработки программного обеспечения количество объектов будет увеличиваться.



Программная инженерия направлена на изучение методов и подходов к построению программного обеспечения, которое можно применять в бесчисленных сферах, таких как бизнес, образование, социальный сектор и другие.

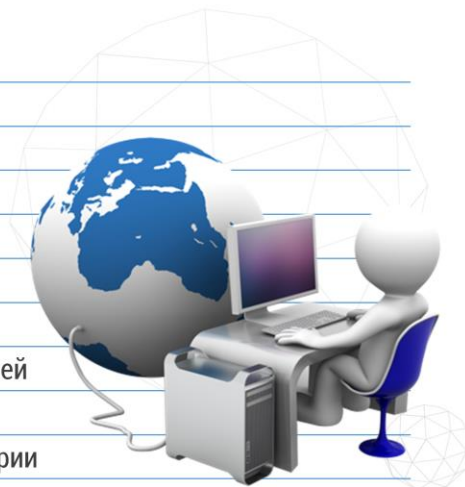
Программное обеспечение разработано для решения некоторых конкретных задач, таких как обработка данных, обмен информацией, общение и так далее. Оно классифицируется по диапазону возможностей применения. Эти классификации перечислены ниже.

- Системное программное обеспечение: этот класс программного обеспечения управляет и контролирует внутренние операции компьютерной системы. Это группа программ, которая отвечает за эффективное и действенное использование ресурсов компьютера. Например, операционная система.
- Программное обеспечение реального времени: этот класс программного обеспечения наблюдает, анализирует и контролирует реальные события в мире по мере их возникновения. Например, программное обеспечение, используемое для прогнозирования погоды.
- Программное обеспечение для бизнеса: этот класс программного обеспечения широко используется в областях, где управление и контроль финансовой деятельности имеют первостепенное значение.

- Инженерное и научное программное обеспечение: этот класс программного обеспечения стал мощным инструментом в исследованиях и разработках технологий следующего поколения.
- Программное обеспечение искусственного интеллекта: этот класс программного обеспечения используется там, где метод решения проблем не является алгоритмическим по своей природе. Для решения этих проблем требуются конкретные стратегии, которые включают экспертную систему, распознавание образов и игровые техники.
- Программное обеспечение на основе Интернета: этот класс программного обеспечения действует как интерфейс между пользователем и Интернетом.
- Программное обеспечение для персонального компьютера: этот класс программного обеспечения используется как в служебных, так и в личных целях. Это программное обеспечение используется преимущественно почти во всех областях.

СУБДИСЦИПЛИНЫ ПО

Требования к ПО
Проектирование ПО
Разработка ПО
Тестирование ПО
Обслуживание ПО
Управление конфигурацией ПО
Управление программной инженерией
Процесс разработки ПО
Инструменты программной инженерии
Качество ПО



Программную инженерию можно разделить на десять субдисциплин.

1. Требования к программному обеспечению: выявление, анализ, спецификация и проверка требований к программному обеспечению.
2. Проектирование программного обеспечения: обычно выполняется с помощью инструментов компьютерной инженерии программного обеспечения и использует стандарты для формата, такие как унифицированный язык моделирования UML .
3. Разработка программного обеспечения: создание программного обеспечения с использованием языков программирования.
4. Тестирование программного обеспечения.
5. Обслуживание программного обеспечения: программные системы часто имеют проблемы и нуждаются в доработках в течение длительного времени после их первого завершения.
6. Управление конфигурацией программного обеспечения: поскольку программные системы очень сложны, их конфигурация должна управляться стандартизированным и структурированным методом. Например, управление версиями и исходным кодом.
7. Управление программной инженерией: управление программными системами во многом заимствует из управления проектами, но есть нюансы,

встречающиеся в программном обеспечении, которых нет в других дисциплинах управления.

8. Процесс разработки программного обеспечения: процесс создания программного обеспечения горячо обсуждается среди практиков, причем основными парадигмами являются гибкая разработка или водопад.
9. Инструменты программной инженерии.
10. Качество программного обеспечения.

Более подробно мы потом рассмотрим каждую из выделенных областей. А также в процессе обучения вы практически сможете посмотреть, какие знания нужны будут вам, чтобы вы смогли освоить определенные навыки по разработке программного продукта.

ФУНКЦИИ ПРОГРАММНОЙ ИНЖЕНЕРИИ

изучить методы и средства разработки компьютерных программ

применить накопленный опыт прикладного программирования к разработке ПО

определить комплекс автоматизированных операций для разработки ПО

определить свод правил для разработки сложных объектов как совокупности простых

формализовать методы оценки готовых программных продуктов

Производственная дисциплина

Инженерная дисциплина

Дисциплина управления

Экономическая дисциплина



Программная инженерия рассматривается как совокупность процесса разработки программ и индустриального подхода. Это отложило определенный отпечаток и позволило сформировать перечень следующих функций, которые выполняет программная инженерия.

- Изучение методов и средств построения компьютерных программ.
- Отображение закономерностей развития и обобщение накопленного опыта прикладного программирования.
- Определение автоматизированных операций по производству объектов: модулей, компонентов, программных аспектов и тому подобное.
- Определение правил и порядка инженерной деятельности для построения из простых объектов новых, более сложных программных систем, семейств систем, проектов и тому подобное.
- Формализация методов измерения и оценки готовых программных продуктов.

Программная инженерия, исходя из перечисленных функций, может быть рассмотрена в таких аспектах, как:

- производственная дисциплина;
- инженерная дисциплина;
- дисциплина управления;

- экономическая дисциплина.

Рассмотрим теперь программную инженерию в соответствии с выделенными аспектами.

ПРОГРАММНАЯ ИНЖЕНЕРИЯ КАК НАУКА

Цель программной инженерии как науки: использование знаний для построения программных продуктов

Фундаментальные науки

алгоритмы

мат логика

управление

доказательства

множества

Программная инженерия как наука

основные понятия и объекты

программирование

разработка ПО

инструментарий разработчика



Цель программной инженерии как науки: применение знаний, полученных из фундаментальных наук, для построения сложных программных продуктов. Построение программного продукта включает анализ автоматизируемой предметной области и производство кода для выполнения на компьютере. Фундаментальные науки, используемые в программной инженерии, это:

- теория алгоритмов;
- математическая логика;
- теория управления;
- теория доказательств;
- теория множеств.

Программная инженерия как наука содержит в себе:

- основные понятия и объекты;
- теорию программирования;
- методы управления изготовлением программного обеспечения;
- средства и инструменты процессов изготовления.

Кроме того, в настоящее время повышение важности исследовательского программного обеспечения мотивирует необходимость определить и понять, какие методы программной инженерии подходят. Из-за уникальности области исследовательского программного обеспечения существующие инструменты и

методы программной инженерии, разработанные для бизнеса или ИТ-сообщества, часто неэффективны. Соответствующие решения программной инженерии должны учитывать характерные особенности среды разработки программного обеспечения для исследований. Чтобы найти эти решения, члены сообщества программной инженерии должны взаимодействовать с членами сообщества исследовательского программного обеспечения.

ПРОГРАММНАЯ ИНЖЕНЕРИЯ КАК ПРОИЗВОДСТВЕННАЯ ДИСЦИПЛИНА

Цель программной инженерии как производственной дисциплины: разработка программных продуктов с применением готовых решений и информационных ресурсов сети Интернет

Инженерные подходы к применению программных продуктов:

повторное использование кода

инженерия приложений

инженерия предметной области



Цель программной инженерии как производственной дисциплины: изготовление программных продуктов с использованием доступных разработанных готовых программ и информационных ресурсов сети Интернет. Программные продукты строятся из компонентов повторного использования, которые хранятся в свободном доступе в репозиториях программного обеспечения.

Инженерные подходы к применению программных продуктов:

- повторное использование кода: проектирование системы снизу вверх;
- инженерия приложений: проектирование системы сверху вниз;
- инженерия предметной области: одновременное проектирование семейства программных продуктов; напоминает процесс конвейерной сборки.

Рассмотрим чуть более подробно выделенные инженерные подходы.

1. Повторное использование кода – это использование существующего программного обеспечения или знаний о программном обеспечении для создания нового программного обеспечения. Повторное использование кода направлено на экономию времени и ресурсов и сокращение избыточности за счет использования ранее созданных активов. Ключевая идея повторного использования состоит в том, что части компьютерной программы могут или должны использоваться при построении других программ, написанных позже.

2. Разработка приложений – это процесс синтеза для создания и поддержки прикладного продукта, который удовлетворяет заданным требованиям заказчика. Разработка приложений характеризуется комплексным процессом жизненного цикла продукта как набора рабочих продуктов, которые предлагают возможности для повторного использования.

3. Разработка предметной области представляет собой процесс повторного использования знаний предметной области при производстве новых программных систем. Ключевой идеей систематического повторного использования программного обеспечения является домен. Вместо того, чтобы создавать каждый новый вариант системы с нуля, можно добиться значительной экономии за счет повторного использования частей предыдущих систем в домене для создания новых.

ПРОГРАММНАЯ ИНЖЕНЕРИЯ КАК ДИСЦИПЛИНА УПРАВЛЕНИЯ

Цель программной инженерии как дисциплины управления:
автоматизация процесса управления командной разработкой
программного продукта

Ядро знаний PMBOK

управление содержанием проекта

управление качеством

управление человеческими
ресурсами



 **Росдистант**
ВЫСШЕЕ ОБРАЗОВАНИЕ ДИСТАНЦИОННО

Цель программной инженерии как дисциплины управления: автоматизация и оптимизация управления процессами разработки программного продукта. Теоретическая основа дисциплины управления – это теория управления сложными системами, разработанная в 1970-х годах Виктором Михайловичем Глушковым.

Стандарты управления собраны в ядре знаний PMBOK и стандарте IEEE Std.1490 и включают следующее.

- Управление содержанием проекта: процессы, необходимые для выполнения работ по проекту и его планирования с расщеплением работ на более простые для упрощения процесса управления.
- Управление качеством: процессы, связанные с обеспечением качества согласно заданным условиям и контролем качества конечного продукта.
- Управление человеческими ресурсами: организация и распределение работ между исполнителями согласно их квалификации.

Свод знаний по управлению проектами PMBOK – это документ, содержащий стандартную терминологию, передовой опыт и руководящие принципы процесса управления проектами. Большая часть руководства охватывает темы, относящиеся к управлению проектами. Но упоминает и более широкие темы, которые частично совпадают, такие как бюджетирование, планирование,

финансовое прогнозирование, укомплектование персоналом, организационное поведение и наука об управлении. Согласно руководству, три фазы проекта включают входы, инструменты, методы и выходы. Они используются для создания общего жизненного цикла проекта с описанием типичных шагов и методов управления проектом.

ЭКОНОМИЧЕСКАЯ ДИСЦИПЛИНА ПРОГРАММНОЙ ИНЖЕНЕРИИ

Цель экономики программной инженерии:

- оценка показателей разработки ПО
- определение рисков разработки ПО при заданных ресурсах

Математические модели:

COCOMO

ANGEL

SLIM

и так далее



Цель экономики программной инженерии заключается в следующем:

- оценка ценовых, временных и экспертных показателей для составления контрактов на создание программного продукта, принятия проектных решений, разработки архитектуры и тому подобное;
- определение рисков проектирования при заданных ресурсах.

Для оценки затрат на производство программных продуктов используются математические модели: COCOMO, ANGEL, SLIM и так далее.

Наиболее широко применяется модель COCOMO – конструктивная модель затрат. Это регрессионная модель, основанная на количестве строк кода. Это процедурная модель оценки стоимости программных проектов, которая часто используется как процесс надежного прогнозирования различных параметров, связанных с созданием проекта, таких как размер, стоимость, время и качество. Таким образом, основными составляющими программной инженерии являются ее научное, инженерное и производственное направления, а также дисциплины управления и экономики.

Процессы программной инженерии описаны в базах знаний SWEBOK и PMBOK и в стандартах ISO.

СТАНДАРТЫ ПРОГРАММНОЙ ИНЖЕНЕРИИ

ISO 12207 - международный стандарт разработки программного обеспечения

Описывает процессы жизненного цикла программного обеспечения

Основные стандарты:

ISO/IEC 12207 «Процессы жизненного цикла программного обеспечения»

ISO/IEC 14598 «Оценивание программного продукта»

ISO 15939 «Процесс измерения»

ISO/IEC 15504 «Оценивание процессов жизненного цикла ПО»

ISO 9001 «Системы управления качеством»

ISO/IEC TR 9126 «Программная инженерия. Качество продукта»



Основная часть стандартов, регламентирующих программную инженерию, – это ISO 12207. Это международный стандарт программной инженерии, который определяет процесс разработки программного обеспечения, действия и задачи – все, что связано с процессом жизненного цикла программного обеспечения от начала до вывода из эксплуатации.

Основная цель стандарта – предоставить общую структуру, чтобы все, кто участвует в разработке программного обеспечения, использовали общий язык. Он стремится быть стандартом, определяющим все задачи, необходимые для разработки и сопровождения программного обеспечения.

Основные стандарты, касающиеся программной инженерии, следующие.

- ISO/IEC 12207 – «Процессы жизненного цикла программного обеспечения»;
- ISO/IEC 14598 – «Оценивание программного продукта»;
- ISO 15939 – «Процесс измерения»;
- ISO/IEC 15504 – «Оценивание процессов жизненного цикла ПО»;
- ISO 9001 – «Системы управления качеством»;
- ISO/IEC TR 9126 – «Программная инженерия. Качество продукта».

Кроме перечисленных стандартов, которые используются при разработке программного обеспечения, в программной инженерии используется кодекс.

КОДЕКС ЭТИКИ И ПРОФЕССИОНАЛЬНОЙ ПРАКТИКИ РАЗРАБОТКИ ПО

Общественность

Клиент и работодатель

Продукт

Суждение

Управление

Профессия

Сотрудники

Самостоятельность



Кодекс этики и профессиональной практики разработки программного обеспечения рекомендован Объединенной целевой группой ACM/IEEE-CS в качестве стандарта для обучения и применения программного обеспечения. Согласно данному кодексу, программные инженеры обязаны сделать анализ, спецификацию, проектирование, разработку, тестирование и сопровождение программного обеспечения полезной и уважаемой профессией. В соответствии со своими обязательствами по охране здоровья, безопасности и благополучия общества инженеры-программисты должны придерживаться следующих восьми принципов.

Первый принцип – ОБЩЕСТВЕННОСТЬ. Разработчики программного обеспечения должны действовать в соответствии с общественными интересами.

Второй принцип – КЛИЕНТ И РАБОТОДАТЕЛЬ. Инженеры-программисты должны действовать таким образом, который отвечает интересам их клиента и работодателя в соответствии с общественными интересами.

Третий принцип – ПРОДУКТ. Инженеры-программисты должны гарантировать, что их продукты и связанные с ними модификации соответствуют самым высоким профессиональным стандартам.

Четвертый принцип – СУЖДЕНИЕ. Программные инженеры должны сохранять честность и независимость в своих профессиональных суждениях.

Пятый принцип – УПРАВЛЕНИЕ. Менеджеры и руководители программной инженерии должны придерживаться этического подхода к управлению разработкой и сопровождением программного обеспечения и продвигать его.

Шестой принцип – ПРОФЕССИЯ. Инженеры-программисты должны поддерживать честность и репутацию своей профессии в соответствии с общественными интересами.

Седьмой принцип – СОТРУДНИКИ. Инженеры-программисты должны быть справедливыми по отношению к своим коллегам и поддерживать их.

И последний восьмой принцип – САМОСТОЯТЕЛЬНОСТЬ. Инженеры-программисты должны участвовать в непрерывном обучении своей профессии и придерживаться этического подхода к профессиональной деятельности.

ПРИНЦИПЫ КОДЕКСА

Принципы поведения программистов и принципы их решений

Принципы этических отношений в коллективе

Принципы обязательства между членами команды разработчиков

Принципы должны:

- определить круг лиц, на которых влияют;
- сформировать уважительное отношение друг к другу;
- оценить мнение общественности к решениям при их информировании;
- проанализировать степень влияния решения на уполномоченных;
- оценить последствия принятых решений инженера-программиста.



Выделенные принципы связаны с поведением профессиональных инженеров-программистов и принимаемыми ими решениями, включая практиков, преподавателей, менеджеров, руководителей и политиков, а также стажеров и студентов данной профессии. Принципы определяют этически ответственные отношения, в которых участвуют отдельные лица, группы и организации, а также основные обязательства в рамках этих отношений.

Пункты каждого принципа иллюстрируют некоторые обязательства, включенные в эти отношения. Эти обязательства основаны на человечности программиста, особой заботе о людях, на которых влияет работа разработчиков программного обеспечения, и уникальных элементах практики разработки программного обеспечения. Кодекс предписывает это в качестве обязательств для любого, кто заявляет о себе или стремится стать разработчиком программного обеспечения. Эти принципы должны побуждать разработчиков программного обеспечения:

- рассмотреть того, на кого влияет их работа;
- проверить, относятся ли они и их коллеги к другим людям с должным уважением;
- рассмотреть, как общественность, если она достаточно хорошо информирована, отнесется к их решениям.
- Кроме того, проанализировать, как их решения повлияют на наименее

уполномоченных;

- рассмотреть, будут ли их действия признаны достойными идеального профессионала, работающего в качестве инженера-программиста.

Кодекс – это не простой этический алгоритм, который позволяет принимать этические решения. Кодекс предназначен не просто для определения характера сомнительных действий; он также выполняет важную образовательную функцию. Поскольку этот кодекс выражает консенсус профессионалов по этическим вопросам, он является средством информирования как общественности, так и начинающих профессионалов об этических обязательствах всех инженеров-программистов.

ПРИНЦИП 1: ОБЩЕСТВЕННЫЙ

Принимать на себя полную ответственность

Соблюдать умеренность интересов

Одобрять программное обеспечение

Сообщать о рисках

Сотрудничать

Быть честным и избегать обмана

Рассматривать проблемы программного обеспечения

Поощрять использование профессиональных навыков



Программные инженеры должны действовать в соответствии с общественными интересами. В частности, инженеры-программисты должны при необходимости осуществлять следующее.

- Принимать на себя полную ответственность за свою работу.
- Соблюдать умеренность интересов разработчика программного обеспечения, работодателя, клиента и пользователей в интересах общества.
- Одобрять программное обеспечение только в том случае, если есть веские основания полагать, что оно безопасно, соответствует спецификациям и тому подобное. Конечный результат работы должен приносить пользу обществу.
- Сообщать соответствующим лицам или властям о любой фактической или потенциальной опасности для пользователя, общественности или окружающей среды, если это связано с программным обеспечением или соответствующими документами.
- Сотрудничать в решении вопросов, вызывающих серьезную общественную озабоченность, относящихся к программному обеспечению, его установке, обслуживанию, поддержке или документации.
- Быть честными и избегать обмана во всех заявлениях, особенно публичных, относительно программного обеспечения или связанных с ним документов, методов и инструментов.

- Рассматривать проблемы физических недостатков, распределения ресурсов, экономического неблагополучия и других факторов, которые могут ограничить доступ к преимуществам программного обеспечения.
- Поощрять добровольное использование профессиональных навыков для добрых дел и вносить свой вклад в общественное просвещение.

ПРИНЦИП 2: КЛИЕНТ И РАБОТОДАТЕЛЬ

Предоставлять услуги

Не использовать программное обеспечение, полученное незаконным образом

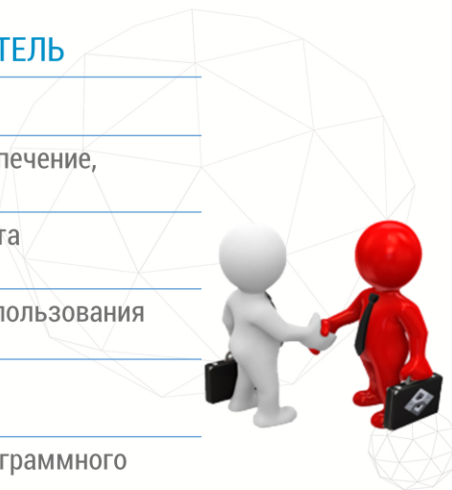
Использовать собственность клиента разрешенным способом

Убедиться, что в правомерности использования документом

Соблюдать коммерческую тайну о конфиденциальной информации

Сообщать о проблемах в работе программного обеспечения

Представлять интересы клиентов



Программные инженеры должны действовать в соответствии с интересами своего клиента и работодателя и в соответствии с общественными интересами. В частности, инженеры-программисты должны при необходимости осуществлять следующее.

- Предоставлять услуги в областях своей компетенции, честно и открыто заявляя о любых ограничениях своего опыта и образования.
- Не использовать сознательно программное обеспечение, полученное или сохраненное незаконным или неэтичным образом.
- Использовать собственность клиента или работодателя только надлежащим образом разрешенным способом, с ведома и согласия клиента или работодателя.
- Убедиться, что любой документ, на который они полагаются, был утвержден.
- Хранить в тайне любую конфиденциальную информацию, полученную в ходе их профессиональной деятельности, если такая конфиденциальность соответствует общественным интересам и не противоречит закону.
- Выявить, задокументировать, собрать доказательства и незамедлительно сообщить клиенту или работодателю, если проект может потерпеть неудачу.
- Выявить, задокументировать и сообщить работодателю или клиенту о важных социальных проблемах в программном обеспечении или связанных

документах.

- Не принимать внешнюю работу, наносящую ущерб работе, которую они выполняют для своего основного работодателя.
- Не продвигать интересы, неблагоприятные для работодателя или клиента.

ПРИНЦИП 3: ПРОДУКТ

Стремится к высокому качеству

Обеспечивать цели и задачи программного продукта

Соотнести требования и квалификационные навыки

Определить требуемый метод разработки

Соблюдать профессиональные стандарты

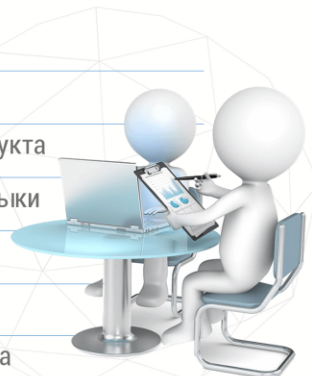
Описывать требования и спецификации продукта

Задokumentировать спецификации программного обеспечения

Дать качественную и количественную оценку продукту

Протестировать продукт и осуществить отладку

Задokumentировать процесс разработки и сформировать пользовательские инструкции



Росдистант
ВЫСШЕЕ ОБРАЗОВАНИЕ ДИСТАНЦИОННО

Программные инженеры должны гарантировать, что их продукты и связанные с ними модификации соответствуют самым высоким профессиональным стандартам. В частности, инженеры-программисты должны при необходимости выполнять следующее.

- Стремиться к высокому качеству, приемлемой стоимости и разумному графику.
- Обеспечить правильные и достижимые цели и задачи для любого проекта, над которым они работают или предлагают.
- Убедиться, что обладают квалификацией для любого проекта, над которым работают или предлагают работать.
- Убедиться, что для любого проекта, над которым они работают или предполагают работать, используется соответствующий метод.
- Работать в соответствии с профессиональными стандартами, если таковые имеются, которые наиболее подходят для поставленной задачи.
- Стремиться полностью понять спецификации программного обеспечения, с которым они работают.
- Убедиться, что спецификации программного обеспечения, над которым они работают, хорошо задokumentированы, удовлетворяют требованиям пользователей и имеют соответствующие утверждения.

- Обеспечить реалистичные количественные оценки затрат, графиков, персонала, качества и результатов по любому проекту, над которым они работают.
- Обеспечить адекватное тестирование, отладку и проверку программного обеспечения и связанных документов, с которыми они работают.
- Обеспечить адекватную документацию, включая обнаруженные существенные проблемы и принятые решения, для любого проекта, над которым они работают.
- Использовать только точные данные, полученные этичным и законным способом, и реализовывать их только разрешенными способами.
- Поддерживать целостность данных.

ПРИНЦИП 4: ПРИГОВОР

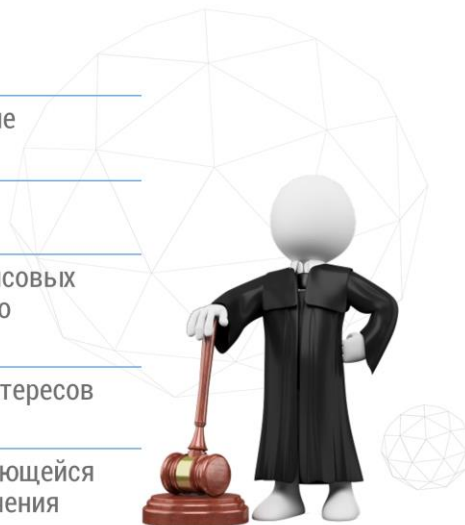
Утверждать документы, прошедшие контроль

Быть объективными при оценке программного обеспечения

Давать объективную оценку финансовых затрат на разработку программного обеспечения

При возникновении конфликтов интересов информировать все участников

Не вступать в организации, занимающейся проблемами программного обеспечения



Программные инженеры должны сохранять честность и независимость в своих профессиональных суждениях. В частности, инженеры-программисты должны при необходимости осуществлять следующее.

- Подтверждать только документы, подготовленные под их контролем или в пределах их компетенции и с которыми они согласны.
- Сохранять профессиональную объективность в отношении любого программного обеспечения или связанных документов, которые их просят оценить.
- Не участвовать в финансовых методах, вводящих в заблуждение, таких как взятничество, двойное выставление счетов или другие неправомерные финансовые методы.
- Сообщать всем заинтересованным сторонам о конфликтах интересов, которых невозможно избежать.
- Отказаться от участия в качестве членов или консультантов в частной, государственной или профессиональной организации, занимающейся проблемами программного обеспечения.

ПРИНЦИП 5: УПРАВЛЕНИЕ

Обеспечить управление проектом разработки ПО

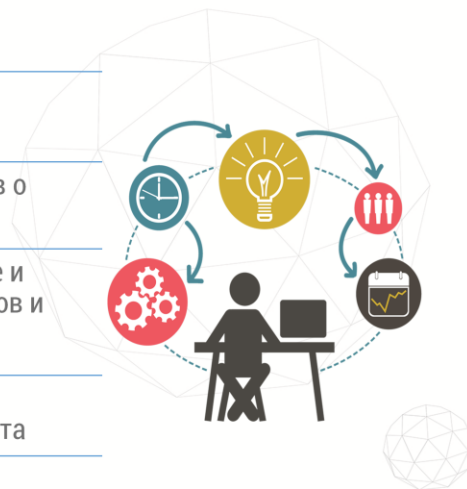
Проинформировать программистов о стандартах

Оценить уровень знаний о политике и процедурах защиты паролей, файлов и информации

Выбрать подходящую методику количественного оценивания проекта

Сформировать команду профессиональных разработчиков

Обеспечить правовую процедуру сопровождения разработки ПО



Руководители программной инженерии должны придерживаться этического подхода к управлению разработкой и сопровождению программного обеспечения и продвигать его. В частности, руководители или ведущие инженеры-программисты должны при необходимости предпринимать следующее.

- Обеспечить хорошее управление любым проектом, над которым они работают, включая эффективные процедуры для повышения качества и снижения риска.
- Убедиться, что инженеры-программисты проинформированы о стандартах, прежде чем будут им следовать.
- Убедиться, что знают политику и процедуры работодателя по защите паролей, файлов и информации, которая является конфиденциальной для работодателя или конфиденциальной для других.
- Обеспечить реалистичные количественные оценки затрат, расписания, персонала, качества и результатов по любому проекту, над которым они работают или предполагают работать.
- Привлечь потенциальных программистов только полным и точным описанием условий трудоустройства.
- Обеспечить надлежащую правовую процедуру при рассмотрении обвинений

в нарушении политики работодателя или настоящего кодекса.

ПРИНЦИП 6: ПРОФЕССИЯ

Создать организационную среду взаимодействия

Сформировать свод знаний в области разработки программного обеспечения

Соблюдать интересы командной разработки

Соблюдать стандарты и законы программной инженерии

Точно описывать спецификации и характеристики программного обеспечения

Отвечать за допущенные ошибки

Избегать ассоциативного подхода при разработке ПО

Сообщать о серьезных ошибках и рисках



Программные инженеры должны быть честными и поддерживать репутацию своей профессии в соответствии с общественными интересами. В частности, инженеры-программисты должны при необходимости осуществлять следующее.

- Помочь создать организационную среду, благоприятную для этического поведения.
- Расширить знания в области разработки программного обеспечения, участвуя в профессиональных организациях, встречах и публикациях.
- Не продвигать собственные интересы за счет профессии, клиента или работодателя.
- Соблюдать все законы, регулирующие их работу, кроме случаев исключительных обстоятельств, когда такое соблюдение несовместимо с общественными интересами.
- Быть точными в описании характеристик программного обеспечения.
- Взять на себя ответственность за обнаружение, исправление и сообщение об ошибках в программном обеспечении и связанных документах.
- Избегать ассоциаций с предприятиями и организациями, которые противоречат кодексу.
- Признать, что нарушения этого кодекса несовместимы с деятельностью

профессионального программиста.

- Сообщать о серьезных нарушениях настоящего кодекса в соответствующие органы.

ПРИНЦИП 7: КОЛЛЕГИ



Соблюдать Кодекс самому и коллективом

Содействовать в профессиональном развитии

Формировать доверие в коллективе

Анализировать работу, не принимая субъективные решения

Выслушивать мнения и принимать объективные решения

Оказывать поддержку коллегам

Помогать профессиональному росту коллеги



Программные инженеры должны справедливо относиться к своим коллегам и поддерживать их. В частности, инженеры-программисты должны при необходимости совершать следующее.

- Призывать коллег соблюдать этот кодекс.
- Содействовать коллегам в профессиональном развитии.
- Полностью доверять чужой работе и воздерживаться от чрезмерного признания.
- Объективно, откровенно и задокументированно анализировать работу других.
- Беспристрастно выслушивать мнения, опасения или жалобы коллеги.
- Помогать коллегам полностью ознакомиться с текущими стандартными методами работы, включая политику и процедуры защиты паролей, файлов и другой конфиденциальной информации.
- Не вмешиваться несправедливо в карьеру коллеги.
- В ситуациях, выходящих за рамки их собственных компетенций, обращаться к мнению других профессионалов, обладающих компетенцией в этой области.

ПРИНЦИП 8: САМОСТОЯТЕЛЬНОСТЬ

Повышать профессиональные знания в области разработки и сопровождения программного обеспечения

Повышать профессиональные навыки в области разработки безопасного, надежного и полезного качественного программного обеспечения

Повышать навыки грамотного документирования процесса разработки программного обеспечения

Повышать навыки тестирования и оценки программного обеспечения

Повышать знания стандартов и законов, регулирующих программное обеспечение

Иметь собственное решение по разработке программного обеспечения

Не нарушать Кодекс



Инженеры-программисты должны участвовать в непрерывном обучении своей профессии и продвигать этический подход в своей работе. В частности, инженеры-программисты должны постоянно стремиться к следующему.

- Усовершенствовать свои знания о разработках в области анализа, спецификации, проектирования, разработки, сопровождения и тестирования программного обеспечения и связанных документов, а также управления процессом разработки.
- Повышать способность создавать безопасное, надежное и полезное качественное программное обеспечение по разумной цене и в разумные сроки.
- Повышать способность создавать точную, информативную и хорошо написанную документацию.
- Улучшать понимание программного обеспечения и связанных документов, с которыми они работают, а также среды, в которой они будут использоваться.
- Улучшать знания соответствующих стандартов и законов, регулирующих программное обеспечение, и сопутствующие документы, с которыми они работают.
- Не оказывать влияние на других с целью совершения каких-либо действий, связанных с нарушением настоящего кодекса.

- Признать, что личные нарушения кодекса несовместимы с деятельностью профессионального программиста.

Этот кодекс был разработан совместной рабочей группой ACM/IEEE-CS по этике и профессиональной практике программной инженерии.

Мы рассмотрели основные стандарты и принципы кодекса, а теперь должны ознакомиться со сводом знаний программной инженерии.