

ЭТАПЫ РАЗРАБОТКИ ПРОГРАММНОГО ПРОДУКТА

М.С. Милованова, студентка группы 17В30,

научный руководитель: Чернышева Т.Ю., к.т.н., доц.

Юргинский технологический институт (филиал) Национального исследовательского

Томского политехнического университета

652055, Кемеровская обл., г. Юрга, ул. Ленинградская, 26

Любой программный продукт перед окончательной реализацией проходит несколько главных фаз. Нюансы разработки могут быть разными для каждой отдельно взятой программы, но основная последовательность всегда остается неизменной. В программировании такими этапами являются:

- Постановка задачи;
- Анализ и моделирование;
- Выбор оптимального решения;
- Написание кода;
- Тестирование;
- Отладка;
- Документирование;
- Публикация.

Началом разработки каждой программы является постановка технического задания. Точность и грамотность тех требований, что были выставлены к проекту, влияет на итоговый результат всей работы. Правильно сформулированное техническое задание должно описывать действия программного продукта, а также те результаты, что от него ожидаются. Для получения точных сведений необходимо ориентироваться на аудиторию возможных пользователей и цель разработки программы. Если оператор может решить задачу аналитически, нет смысла привлекать вычислительные мощности компьютера.

Анализ и моделирование. Когда примерный план разработки программы уже обрел свои очертания, аналитик определяет исходные данные и итоговый результат. В программировании математической моделью считается система математических соотношений, что приближенно отражает сформулированную задачу. Такая модель позволяет выбирать оптимальные варианты решений относительно выбранных критериев. Также при этом выявляются ограничения на использование данных, и создается формальное описание текущих задач.

Разработка алгоритма решения. Научным алгоритмом является определенная последовательность действий, строгое и последовательное исполнение которых позволяет решить поставленную перед алгоритмом задачу за конечное число шагов. При разработке любой программы должны соблюдаться два условия: она должна быть понятна оператору, ее разрабатывающему, и объекту, выполняющему предписанные программой функции. Также у алгоритма любой программы есть пять главных свойств:

- Дискретность (набор конкретных действий, что следуют в определенном порядке);
- Детерминированность (строгое и недвусмысленное определение всех действий в каждом случае);
- Конечность (возможность завершения каждого шага и последовательности алгоритма);
- Массовость (использование одного алгоритма с разными входными значениями);
- Результативность (отсутствие ошибок исполнения и получение правильных данных при всех допустимых входных значениях).

После того как программа проходит начальные этапы, а поставленную задачу всесторонне анализируют, программист приступает к написанию кода.

Написание кода. Кодирование – это процесс написания компьютерного кода, а также подготовка действующих скриптов с целью реализации выбранного алгоритма на определенном языке программирования. Чтобы данный текст был понятен как пользователям, так и составителям, в нем используются комментарии. Главное назначение операции кодирования – это преобразование информации в набор символов, понятых электронно-вычислительной машине. Структурированные таким образом данные будут более компактны и пригодны для передачи. Принципиальная схема информационной обработки состоит из ее поиска, анализа и упорядочения.

Тестирование. Понятие тестирования включает в себя динамический контроль, то есть проверку правильности программы вовремя ее непосредственного исполнения на компьютере. Важность

своевременного тестирования очень высока. В крупных компаниях, занимающихся разработкой программного обеспечения, на тесты и отладку приходится больше половины бюджета. Главная трудность процесса тестирования состоит в его деструктивности. Проще говоря, при неудачном тестировании программа работает нормально, но возможных ошибок выявить не удастся. И наоборот, если выполнение тестируемой программы завершается с ошибкой, тест считается удачным. Главные принципы организации тестирования:

- Описание ожидаемого результата работы программы с целью выявления возможных ошибок;
- Изучение результатов теста на предмет малозаметных неточностей;
- Использование разных видов данных для проверки их взаимодействия с программой;
- Сохранение результатов предыдущих тестов с целью повышения эффективности следующих;
- Принцип скопления ошибок – количество необнаруженных ошибок в элементе программы будет прямо пропорционально количеству ошибок, что уже в нем обнаружены.

Виды тестирования. Динамический контроль программы ведется путем функционального и структурного тестирования. В первом случае тексты программ не используются, проводится только проверка их соответствия внешней спецификации. Поскольку критерием полноты тестирования в этом варианте является перебор всех значений входных данных, на практике это невыполнимо. При структурном тестировании применяются тесты, с большой долей вероятности позволяющие найти ошибку. Поскольку даже для программ средней сложности число таких путей составляет десятки тысяч, такое тестирование тоже не может быть исчерпывающим. По этой причине в тестировании программ, состоящих из нескольких модулей, выделяют четыре главных этапа:

- Тестирование каждого отдельного модуля;
- Общее тестирование программных модулей;
- Проверка функций программных комплексов (поиск различий между готовой программой и ее начальной спецификацией);
- Общее тестирование программного комплекса (проверка соответствия готовой программы тем целям, что были указаны в техническом задании).

Отладка программы. В ходе разработки любой программы всегда возникают ошибки. Если факт их существования установлен, то для их выявления и устранения используется операция отладки. Синтаксическая отладка – это исследование текста программы на предмет синтаксических ошибок. Если в программном коде находится какая-то неточность, то транслятор выдаст сообщение о месте ошибки и ее возможном характере. При получении такого сообщения программист должен исправить ошибку и заново повторить трансляцию. Эффективным методом предотвращения синтаксических ошибок может стать использование справочных систем и соблюдение синтаксических правил для описания функций, объектов и методов.

Семантическая отладка. Семантические ошибки, названные ошибками этапа выполнения, происходят, если автор создает функционирующую программу, но ставит перед ней недопустимую цель. Операторы такой программы будут написаны правильно, с соблюдением законов программирования, но при их выполнении произойдет ошибка. Если программа будет пытаться открыть несуществующий файл, делить на ноль или обращаться к давно удаленным данным, тому виной именно семантические ошибки.

Семантической отладкой в таком случае называется процесс поиска, нахождения и исправления тех ошибок, что возникают при неверном указании логических страниц данных. Существует три главных способа программной отладки. Во-первых, это пошаговая отладка каждой программы с заходом в ее подпрограммы. Во-вторых, это пошаговая программная отладка с выполнением подпрограммы в качестве одного оператора. И, в-третьих, это последовательное выполнение программы до точки ее остановки. Во время такой отладки каждый оператор данной программы выполняется по очереди, а те переменные, на которые этот оператор должен был воздействовать, берутся под строгий контроль. Если крупная программа состоит из менее масштабных подпрограмм, и эти подпрограммы уже отлажены, их все можно рассматривать в качестве одного оператора.

Документирование. В ходе сбора документации накапливается широкий спектр описаний, что облегчают создание программы, а также обогащают ее итоговый результат. Такими документами является постановка задачи, проекты деятельности, алгоритмы создания, и все программные компоненты. Документация, что применяется в создании программы, служит средством взаимодействия

для разработчиков, механизмом контроля над процессом разработки, а также способом передачи пользователю тех данных, без которых программа не будет функционировать.

Пользовательская документация должна объяснять клиенту, как он должен действовать, чтобы запустить и использовать необходимую программу. Если созданная программа предполагает хоть какое-то взаимодействие с пользователями, необходимо добавить справочные данные, что будут облегчать их работу. Качество документации сильно влияет на успех самого программного продукта. Без справочной системы, инструкции по запуску, и пособия по установке, клиент не сможет воспользоваться даже самой мощной и функциональной программой, особенно если сталкивается с ней впервые.

Сопроводительная документация описывает программы с точки зрения их разработки. Данный вид документов необходим, если в ходе использования программы клиент планирует изучать особенности ее конструкции. Также сопровождением является выпуск документации по уже готовой и действующей программе с целью ее совершенствования и обновления. В крупных корпорациях, что занимаются выпуском программного обеспечения, выпуск программ и их сопровождение ведутся разными подразделениями работников, и для их успешного взаимодействия применяется как раз сопроводительная документация. От точности, детальности и достоверности тех данных, что представлены в сопроводительном проекте, зависит правильность функционирования всех следующих версий сопровождаемой программы.

Публикация. После получения готового исполняемого файла с данными программы, их можно запустить и еще раз проверить ее работоспособность и полноту функционирования. Если готовый программный продукт правильно действует и выполняет требования, заложенные разработчиком, то он считается работоспособным.

Литература.

1. Dison// [Электронный ресурс]. Режим доступа: http://www.edsd.ru/ru/principy/cikl_razrabotki_po
2. Absolut// [Электронный ресурс]. Режим доступа :http://ab-solut.net/ru/articles/etapi_po/
3. Компьютер Пресс// [Электронный ресурс]. Режим доступа:<http://compress.ru/article.aspx?id=11321>

ИНФОРМАЦИОННАЯ СИСТЕМА СОПРОВОЖДЕНИЯ ДЕЯТЕЛЬНОСТИ МБУ «ЦЕНТР ТЕХНИЧЕСКОГО КОНТРОЛЯ И ОБСЛУЖИВАНИЯ УЧРЕЖДЕНИЙ УПРАВЛЕНИЯ ОБРАЗОВАНИЯ АДМИНИСТРАЦИИ ГОРОДА ЮРГИ»

В.И. Минков, студент группы 3-17890,

научный руководитель: Молнина Е.В.

Юргинский технологический институт (филиал) Национального исследовательского

Томского политехнического университета

652055, Кемеровская обл., г. Юрга, ул. Ленинградская, 26

E-mail:minkov.vasilii@mail.ru

Объектом исследования является процесс деятельности МБУ «ЦТКО УУО» (далее учреждение), осуществляющий своевременный технический контроль и обслуживание учреждений управления образования нашего города. Учреждение возглавляет директор. В его подчинении находятся заместитель. директора по хозяйственной части, начальник гаража, экономист, юрист (рис. 1).



Рис. 1. Организационная структура Центра