*«Экономика программной инженерии». Кейс к практическому занятию 1*

# Анализ существующих альтернативных программных продуктов.

# Исследование рынка программного обеспечения

Программные продукты обладают своеобразием как с точки зрения их разработки, так и в плане их потребления (пользования). В процессе создания программных продуктов по сравнению с созданием традиционных товаров отсутствует производственная часть в силу «электронного представления» компьютерных программ и данных. Программный продукт готов, когда готов «стартовый образец». Если при создании обычного товара на этой стадии только начинается материальное производство его экземпляров, то для программных продуктов оно заменяется тиражированием, никак не влияющим на характеристики самого продукта (здесь может идти разговор лишь о качестве носителя информации с записанным продуктом).

Поведение потребителя здесь также отличается от поведения потребителя на рынках традиционных товаров. Элементы программного обеспечения относятся к продуктам длительного пользования, кроме того, в силу «нематериальности» компьютерные программы подвержены лишь моральному износу. Поэтому программный продукт покупают только однажды, второй раз тому же покупателю его продать нельзя, продаются лишь обновленные и (предположительно) улучшенные версии. Иногда производитель может продавать один экземпляр продукта и отдельно – «лицензии» на нужное количество дополнительных пользователей продукта.

Программное обеспечение можно также распространять через сети передачи данных. Эта возможность позволяет ликвидировать зависимость торговли от географического расположения продавца и покупателя и обеспечить мгновенную доставку продукта потребителю.

Исследования по данной проблематике будут проводиться по схеме, показанной на рисунке 1.

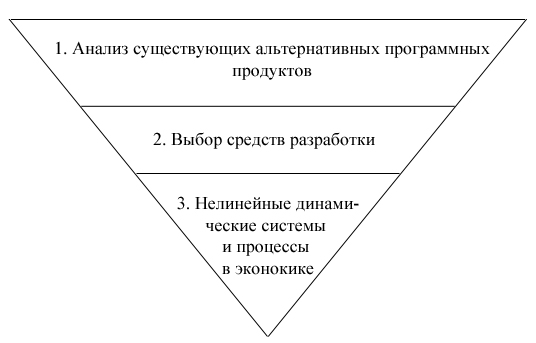


Рисунок 1 – Направление и охват исследований

На начальных этапах разработки любого продукта (в том числе и программного обеспечения), необходимо провести анализ существующих альтернативных продуктов, затем выбрать средства для создания продукта и, в случае данного дипломного проекта, провести анализ нелинейных динамических систем и процессов в экономике, тем самым постепенно сужая охват исследования.

## Анализ существующих альтернативных программных продуктов (на примере)

До начала разработки программного продукта необходимо провести исследование рынка и выяснить, кем и насколько продукт востребован и присутствуют ли другие источники удовлетворения спроса.

Современный информационный рынок предлагает небольшой список научных программных продуктов, разработанных различными фирмами-производителями, позволяющими работать с различными математическими моделями.

Основными примерами подобного программного обеспечения являются следующие программные средства:

1. MATLAB – пакет прикладных программ и язык программирования, используемый для решения задач технических вычислений и компьютерного моделирования на ЭВМ. Разработан и поддерживается компанией «The MathWorks, Inc», США. Является коммерческим приложением;
2. Maple – специализированное программное обеспечение для проведения как аналитического, так и имитационного моделирования. Разработчиком данного программного продукта является канадская компания «[Waterloo Maple, Inc](http://ru.wikipedia.org/w/index.php?title=Waterloo_Maple_Inc.&action=edit&redlink=1)». Также является коммерческим приложением;
3. Mathcad – коммерческая программа для выполнения и документирования инженерных и научных расчётов, работы с математическими моделями. Разработчик – «MathSoft», США. В 2006 году «MathSoft» была поглощена компанией «Parametric Technology Corporation».

Существует также ряд других программных продуктов в области компьютерного моделирования и работы с математическими моделями. Некоторые из них уже давно не поддерживаются производителями. Другие же, в отличие от разрабатываемого ПО идентификации, используют иные модели (в основном, линейные) и методы. Поэтому в данной выпускной квалификационной работе они не были включены в исследование рынка ПО.

Поиск альтернативных программных продуктов производился с помощью опроса специалистов, изучения специальной литературы, а также с помощью поисковых запросов в сети Интернет. Использовались крупнейшие мета-поисковые системы, такие как «Metabot» (<http://www.metabot.ru/>) и «Meta» (<http://meta.ua/> ).

Компании, разрабатывающие вышеперечисленные программные продукты занимают значительную долю рынка программного обеспечения для научных расчетов и обладают большими финансовыми, кадровыми и техническими ресурсами.

В рекламе и продвижении товара они используют методы, традиционные для товарных рынков: устанавливают рекламные щиты, проводят массовые акции, лотереи, конкурсы, участвуют в государственных программах и др. Учитывая специфику научного программного обеспечения и его потребителей, рекламные компании проводятся в научных передачах на телевидении и в прессе, различных конференциях. Кроме того они используют весь арсенал Интернета. Помимо непосредственно собственного сайта в сети Интернет, компании, в том числе стимулируют создание сетевых сообществ пользователей с форумами, тематическими сайтами и базами знаний и применяют мощные системы службы поддержки пользователей.

Распространение программных продуктов осуществляется как непосредственно самими компаниями, так и через посредников, по различным каналам сбыта, представленным на рисунке 2.

Каналы распространения программных продуктов компаниями и посредниками

Почтовая доставка

Сетевой

Традиционный «коробочный»

Рисунок 2 – Схема каналов распространения программных продуктов

Единственным авторизованным партнером компаний «The MathWorks, Inc» и «[Waterloo Maple, Inc](http://ru.wikipedia.org/w/index.php?title=Waterloo_Maple_Inc.&action=edit&redlink=1)» в России является компания «SoftLine» (г. Москва), осуществляющая реализацию их продукции с помощью всех трех каналов распространения программного обеспечения.

Более основательно на российском рынке ПО представлена компания «Parametric Technology Corporation», разрабатывающая Mathcad. Она имеет два собственных офиса в Москве и Санкт-Петербурге. Авторизованными партнерами на территории РФ являются 12 организаций, 10 из которых являются международными. Ведущим партнером является компания «Irisoft» (г. Санкт-Петербург), являющаяся так называемым «платиновым» партнером.

Проведем сравнительный анализ вышеописанных программных средств.

Для реализации компьютерного моделирования в MATLAB используется дополнительно ПО - Simulink, разработанное той же компанией (The MathWorks, Inc). Simulink требуется для любого проектирования и моделирования в среде MATLAB и тесно интегрируется в нее, не являясь самостоятельным ПО. Является расширяемым программным обеспечением за счет использования библиотек готовых блоков. Позволяет работать с различными видами линейных и нелинейных объектов. Библиотека источников сигнала реализует генерацию сигнала, подаваемого на вход модели, а средствами самой MATLAB возможна статистическая обработка результатов моделирования.

В Maple, так же как и в MATLAB для моделирования используется дополнительное ПО – MapleSim. В отличие от традиционных технологий моделирования, оно позволяет просто нарисовать блок-схему работы нужной системы и MapleSim автоматически сгенерирует все уравнения, описывающие работу модели. MapleSim содержит более 500 готовых модельных компонентов для 10 прикладных областей, таких, как электрические сети и устройства, механика, системы управления, тепловые сети и др. Все эти компоненты систематизированы и разбиты на удобные наборы. ПО предлагает пользователям широкий набор аналитических инструментов с поддержкой контроля чувствительности, оптимизацией моделей и расширенной визуализацией сложных моделей из большого числа отдельных деталей, а также инструменты автоматического документирования всех процессов моделирования.

Mathcad – многоплановый программный продукт, включающий в себя множество функций для различных расчетов и средства для построения и работы с моделями. При моделировании динамических процессов и систем, в данном ПО, положительными сторонами являются: наличие большой библиотеки численных методов, создание математических моделей в виде работающих документов, возможность расширения модели путем копирования частей из других документов, открытость документов.

Данные программные продукты рассчитаны на широкий круг потребителей, в большей степени связанных с наукоемкими отраслями. Это и студенты, преподаватели, конструкторы, инженеры, и вузы, крупные корпорации, НИИ и др. Поэтому их создатели стремились сделать ПО универсальным, что, в свою очередь, привело к загруженности функциями, большинство из которых не требуется конкретному пользователю. Разработка специальных версий данных программных продуктов для отдельных групп пользователей позволило отчасти решить эту проблему. В то же время, по результатам независимых исследований применения ПО, пользователи, приобретающие специальные версии программ, полностью удовлетворяющих их потребности, все равно не использовали большую часть функционала программных изделий, т.е. многие функции программ совершенно не нужны пользователям для решения их задач.

В MATLAB и Maple средства моделирования развиты намного сильнее, реализован охват большего класса моделей. Такое преимущество обусловлено применением дополнительного программного обеспечения. Для Mathcad также существуют различные дополнения для моделирования, но они в основном реализованы для узкоспециализированных задач, т.е. разрабатываются для построения модели конкретного объекта.

Несмотря на большое количество функций и мощные средства моделирования ни в одном из данных программных продуктов не реализован метод идентификации НДО типа фильтра Заде с помощью суммы гармонических сигналов. Реализовать метод возможно, но потребуются временные затраты на обучение и материальные вложения для покупки данного программного обеспечения. Стоимостный критерий является одним из основных при принятии решения разработки альтернативного продукта.

В таблице 1 приведена стоимость альтернативных вариантов разрабатываемого программного изделия идентификации НДО по состоянию на 02.04.10, данные взяты с сайта крупнейшего в России ресейлера программного обеспечения компании «SoftLine».

Таблица 1 – Стоимость программного обеспечения

|  |  |
| --- | --- |
| Наименование | Стоимость (руб.) |
| MathWorks MATLAB 7.10 | 91468.75 |
| MathWorks Simulink 7.5 | 137203.13 |
| **Итого решение от MathWorks** | **228671.88** |
| Maplesoft Maple 13 Prof | 58239.98 |
| Maplesoft MapleSim 3 | 92046.83 |
| **Итого решение от Waterloo Maple** | **150286.81** |
| **PTС Mathcad 14** | **43612.30** |

Стоимость приведенных выше программных продуктов включает цену одной стандартной лицензии для нового пользователя программы, то есть правом на использование данной программой обладает лишь одно физическое лицо.

Преимуществом программы, разрабатываемой в рамках выпускной квалификационной работы, является то, что стоимость капитальных затрат на разработку и внедрение, а также стоимость эксплуатационных расходов будут значительно ниже, чем аналогичные показатели при разработке и внедрение данного метода идентификации с помощью альтернативных программных продуктов. Это будет доказано в третьей главе пояснительной записки. Количество функций уменьшиться до строго необходимого для решения задачи минимума. Сам метод и его компьютерная реализация позволит сократить время на проведение эксперимента. Пользовательский интерфейс будет интуитивно понятен, а работнику будет совершенно не нужно обладать знаниями в программировании.

Определенная функциональность программы идентификации и ее целевая направленность, несомненно, сужают круг потребителей данного программного продукта, но в тоже время в полной мере позволяют конкретному пользователю решать его конкретные задачи. ПО рассчитано на студентов, преподавателей, исследователей, и различные компании, которым в своей деятельности приходится работать с идентификацией и НДО.

Первоначальную версию программного обеспечения, возможно, распространять по каналу Интернет бесплатно для апробации идеи и дальнейшего развития продукта. А в будущем, распространять как коммерческое ПО. Определение стоимости программного изделия является весьма сложной задачей и требует отдельного детального рассмотрения. Для расчета стоимости разрабатываемого программного изделия можно будет применить метод ценообразования «Издержки + Прибыль». При использовании данного метода общий объем издержек на разработку программного продукта суммируется с запланированной прибылью, после чего полученная сумма делиться на число экземпляров продукта, которое планируется реализовать.

Итак, в данном подразделе были найдены и проанализированы альтернативные программные продукты, принято решение о разработка программного изделия идентификации НДО IdentNDO v.1.0, выявлены основные преимущества разрабатываемого ПО. Далее, для успешной реализации проекта, необходимо осуществить анализ и выбрать средство разработки.

## Выбор средств разработки

Выбор средств разработки осуществляется на первых стадиях работы над проектом, после определения требований к создаваемому программному продукту. Он, как и анализ существующих альтернативных программных продуктов, является экономическим процессом, в ходе которого выбираются наиболее оптимальные средства разработки для конкретного проекта. Например, по соотношению стоимости средства к необходимым функциям, по времени и стоимости обучения работы со средством разработки и т.д.

Основной составляющей средств разработки является инструментальное программное обеспечение (Software tools) – программное обеспечение, используемое в ходе разработки, корректировки или развития других программ. Сюда входят языки программирования, интегрированные среды разработки, а также различные вспомогательные программы.

Существующие на сегодняшний день языки программирования можно выделить в следующие группы:

1. универсальные языки высокого уровня;
2. специализированные языки разработчика ПО;
3. специализированные языки пользователя;
4. языки низкого уровня.

Для разработки программы идентификации НДО необходим универсальный язык программирования, позволяющий производить научные расчеты с необходимой временными затратами и точностью. Подобными языками являются: C++, Visual Basic, Java, С#, Object Pascal и другие. Лидером по количеству использования в проектах на сегодняшний день является язык C++. Его основные достоинства это:

1. кроссплатформенность, т.е. возможность разрабатывать программы для самых разных платформ и систем;
2. возможность работать на низком уровне с памятью, адресами, портами, что при неосторожном использовании может легко превратиться в недостаток;
3. мощный препроцессор, унаследованный от языка C, но, как и любой другой мощный инструмент, требует осторожного обращения;
4. возможность создания обобщенных алгоритмов для разных типов данных, их специализация и вычисления на этапе компиляции и другие достоинства.

При этом язык C++ обладает рядом существенных недостатков:

1. подключение интерфейса внешнего модуля через препроцессорную ставку заголовочного файла (#include) серьезно замедляет компиляцию при подключении большого количества модулей;
2. недостаток информации о типах данных во время компиляции;
3. сложность для изучения и компиляции;
4. некоторые преобразования типов неинтуитивны. В частности, операция над беззнаковыми и знаковыми числами дает беззнаковый результат.

Для второстепенных проектов часто используется Visual Basic, позволяющий с высокой скоростью создавать приложения с графическим интерфейсом для операционных систем семейства Microsoft Windows, имеющий простой для изучения синтаксис. Также имеется возможность [компиляции](http://ru.wikipedia.org/wiki/%D0%9A%D0%BE%D0%BC%D0%BF%D0%B8%D0%BB%D1%8F%D1%82%D0%BE%D1%80) программы, как в [машинный код](http://ru.wikipedia.org/wiki/%D0%9C%D0%B0%D1%88%D0%B8%D0%BD%D0%BD%D1%8B%D0%B9_%D0%BA%D0%BE%D0%B4), так и в [P-код](http://ru.wikipedia.org/wiki/P-%D0%BA%D0%BE%D0%B4) (по выбору программиста). В режиме [отладки](http://ru.wikipedia.org/wiki/%D0%9E%D1%82%D0%BB%D0%B0%D0%B4%D0%BA%D0%B0) программа всегда (вне зависимости от выбора) компилируется в P-код, что позволяет приостанавливать выполнение программы, вносить значительные изменения в исходный код, а затем продолжать выполнение: полная перекомпиляция и перезапуск программы при этом не требуется. P-код – это концепция аппаратно-независимого исполняемого кода, который также можно назвать байт-код. Недостатками языка являются:

1. поддержка операционных систем только семейства [Windows](http://ru.wikipedia.org/wiki/Windows) и [Mac OS X](http://ru.wikipedia.org/wiki/Mac_OS_X) (Исключение — VB1 for [DOS](http://ru.wikipedia.org/wiki/DOS));
2. отсутствие механизма [наследования реализации](http://ru.wikipedia.org/wiki/%D0%9D%D0%B0%D1%81%D0%BB%D0%B5%D0%B4%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5_%28%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5%29) объектов. Существующее в языке наследование позволяет наследовать только [интерфейсы](http://ru.wikipedia.org/wiki/COM-%D0%B8%D0%BD%D1%82%D0%B5%D1%80%D1%84%D0%B5%D0%B9%D1%81), но не их реализацию;
3. медленная скорость работы, обусловленная тем, что практически все встроенные функции языка реализованы через библиотеку времени исполнения ([runtime library](http://ru.wikipedia.org/wiki/Runtime_library)), которая, в свою очередь, производит много «лишней» работы по проверке и/или преобразованию типов;
4. возможность отключить средства слежения за объявленными [переменными](http://ru.wikipedia.org/wiki/%D0%9F%D0%B5%D1%80%D0%B5%D0%BC%D0%B5%D0%BD%D0%BD%D0%B0%D1%8F), возможность неявного преобразования переменных, наличие [типа данных](http://ru.wikipedia.org/wiki/%D0%A2%D0%B8%D0%BF_%D0%B4%D0%B0%D0%BD%D0%BD%D1%8B%D1%85) «Variant». По мнению критиков, это даёт возможность писать крайне плохой [код](http://ru.wikipedia.org/wiki/%D0%98%D1%81%D1%85%D0%BE%D0%B4%D0%BD%D1%8B%D0%B9_%D0%BA%D0%BE%D0%B4);
5. отсутствие [указателей](http://ru.wikipedia.org/wiki/%D0%A3%D0%BA%D0%B0%D0%B7%D0%B0%D1%82%D0%B5%D0%BB%D1%8C), низкоуровневого доступа к памяти, ассемблерных вставок. Несмотря на то, что [парадигма](http://ru.wikipedia.org/wiki/%D0%9F%D0%B0%D1%80%D0%B0%D0%B4%D0%B8%D0%B3%D0%BC%D0%B0) Visual Basic позволяет среднему VB-[программисту](http://ru.wikipedia.org/wiki/%D0%9F%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%81%D1%82) обходиться без всего этого, перечисленные вещи также нередко становятся объектами [критики](http://ru.wikipedia.org/wiki/%D0%9A%D1%80%D0%B8%D1%82%D0%B8%D0%BA%D0%B0).

Язык Java – объектно-ориентированный язык программирования, разработанный компанией «Sun Microsystem» в 1995 году. Скомпилированные в байт-код программы Java работают только под управлением виртуальной Java-машины (JVM), поэтому они называются приложениями Java. JVM — программа, обрабатывающая байтовый код и передающая инструкции оборудованию как [интерпретатор](http://ru.wikipedia.org/wiki/%D0%98%D0%BD%D1%82%D0%B5%D1%80%D0%BF%D1%80%D0%B5%D1%82%D0%B0%D1%82%D0%BE%D1%80), но с тем отличием, что байтовый код, в отличие от текста, обрабатывается значительно быстрее. Достоинство подобного способа выполнения программ — в полной независимости байт-кода от [операционной системы](http://ru.wikipedia.org/wiki/%D0%9E%D0%BF%D0%B5%D1%80%D0%B0%D1%86%D0%B8%D0%BE%D0%BD%D0%BD%D0%B0%D1%8F_%D1%81%D0%B8%D1%81%D1%82%D0%B5%D0%BC%D0%B0) и [оборудования](http://ru.wikipedia.org/wiki/%D0%90%D0%BF%D0%BF%D0%B0%D1%80%D0%B0%D1%82%D0%BD%D0%B0%D1%8F_%D0%BF%D0%BB%D0%B0%D1%82%D1%84%D0%BE%D1%80%D0%BC%D0%B0), что позволяет выполнять Java-приложения на любом устройстве, для которого существует соответствующая виртуальная машина. Другой важной особенностью технологии Java является гибкая система безопасности благодаря тому, что исполнение программы полностью контролируется виртуальной машиной. Любые операции, которые превышают установленные полномочия программы (например, попытка несанкционированного доступа к данным или соединения с другим компьютером) вызывают немедленное прерывание. Язык гораздо сильнее типизирован по сравнению с C++, т.е. вносит больше ограничений на действия с переменными и величинами различных типов. Наличие готовых шаблонов проектирования (patterns), которые представляют собой многократно используемые решения широко распространенных проблем, позволяют повышать производительность работы программистов. Также одним из достоинств является наличие большого количества профессиональных и бесплатных сред разработки, поддерживающих данный язык.

Часто к недостаткам концепции виртуальной машины относят то, что исполнение байт-кода виртуальной машиной может снижать производительность программ и алгоритмов, реализованных на языке Java. Данное утверждение было справедливо для первых версий виртуальной машины Java, однако в последнее время оно практически потеряло актуальность, благодаря значительному количеству качественных усовершенствований.

Из-за отсутствия хорошего и законченного средства создания форм язык пока остается малопригоден для разработки оконных приложений. В настоящий момент основная область его применения – сетевые технологии: создание клиент-серверных приложений.

C# – также объектно-ориентированный язык программирования. Разработан корпорацией Microsoft в 1998-2001 годах. C# создавался как язык компонентного программирования, и в этом одно из главных достоинств языка, направленное на возможность повторного использования созданных компонентов. Язык является проще и надежнее C++, реализует возможности наследования и универсализации и тесно связан с мощной средой Framework .Net. Благодаря каркасу Framework . Net, ставшему надстройкой над операционной системой, программисты C# получают те же преимущества работы с виртуальной машиной, что и программисты Java. Эффективность кода даже повышается, поскольку исполнительная среда для C# представляет собой компилятор промежуточного языка, в то время как виртуальная Java-машина является интерпретатором байт-кода.

Недостатками C# являются влияние на использование памяти и производительность. Динамическая компиляция с проверкой во время выполнения работает медленнее, чем код написанный на C/C++. Периодически должны выполняться процедуры сборки мусора для восстановления памяти, и возможно даже удаления некоторого кода из кэша кода. Во время сборки мусора все потоки должны приостанавливаться, и это может иметь отрицательное влияние на производительность системы в реальном времени.

Следующий язык программирования, с помощью которого возможно создать необходимое ПО для научных расчетов Object Pascal (Delphi). Язык разработан в фирме [Apple Computer](http://ru.wikipedia.org/wiki/Apple_Computer) в [1986 году](http://ru.wikipedia.org/wiki/1986_%D0%B3%D0%BE%D0%B4) на базе языка Pascal. Позже был значительно доработан фирмой Borland и в официальной документации компании стал упоминаться, как язык Delphi. Является объектно-ориентированным языком. Его основные достоинства:

1. позволяет писать лаконичный и безопасный (за счет строгой типизации) код;
2. компилятор языка обнаруживает большое количество не только синтаксических, но и семантических ошибок;
3. реализована возможность использования ассемблерных вставок в коде, что позволяет существенно сократить время выполнения определенных функций в создаваемой программе;
4. существует возможность использовать библиотеку визуальных компонентов (VCL). VCL – объектно-ориентированная библиотека для разработки программного обеспечения, разработанная компанией [Borland](http://ru.wikipedia.org/wiki/Borland) для поддержки принципов [визуального программирования](http://ru.wikipedia.org/wiki/%D0%92%D0%B8%D0%B7%D1%83%D0%B0%D0%BB%D1%8C%D0%BD%D0%BE%D0%B5_%D0%BF%D1%80%D0%BE%D0%B3%D1%80%D0%B0%D0%BC%D0%BC%D0%B8%D1%80%D0%BE%D0%B2%D0%B0%D0%BD%D0%B8%D0%B5). Она предоставляет огромное количество готовых к использованию компонентов для работы в самых разных областях программирования.

Недостатками языка являются:

1. необходимость в некоторых ситуациях использовать довольно громоздкие конструкции языка, что приводит к снижению читаемости кода. Подробное комментирование кода позволяет компенсировать данный недостаток;
2. слабая поддержка методов работы с оперативной памятью и другими ресурсами компьютера;
3. ограниченное количество сред разработки, поддерживающих данный язык (а фактически – одна);
4. фактически созданные программы не являются кроссплатформенными и ориентированы на платформу x86 и операционные системы семейства Windows.

Проанализировав достоинства и недостатки языков, осуществим выбор языка программирования, с помощью которого будем создавать ПО идентификации НДО. В большей степени выбор языка программирования определяется опытом и знаниями конкретного языка разработчиком, предназначением функциональными особенностями разрабатываемого ПО. С учетом достаточных знаний и опытом практического применения языка Object Pascal (Delphi) в различных проектах, было принято решение использовать именно его для разработки программы в рамках дипломного проектирования. Возможности Delphi и большое количество готовых математических функций позволяют создать необходимое ПО идентификации, а недостатки языка не окажут негативного влияния на реализацию функциональности ПО. В случае, если потребуется уменьшение времени на обработку математических данных в конкретных функциях программы, будет возможность применять ассемблерные вставки и другие методы оптимизации.

После выбора языка программирования необходимо осуществить выбор среды программирования. Интегрированной средой разработки программного обеспечения (ИСРПО) называют систему программных средств, используемую программистами для разработки ПО.

Поскольку для реализации проекта был выбран язык Delphi, выбор среды программирования является однозначным в пользу одноименной ИСРПО, созданной компанией Borland. Borland Delphi не имеет аналогов по качеству и степени полноты поддержки выбранного языка. Существует несколько версий данной среды, разработанных в период с 1995 по 2009 гг., как самой Borland, так компанией Embarcadero, купивший впоследствии среду разработки. Для разрабатываемого проекта нет необходимости использовать самые последние версии среды, поскольку в них добавлена функциональность, которая не потребуется в ходе разработки ПО идентификации. Например, расширенная поддержка сети Интернет, создания клиент-серверных приложений и баз данных. Учитывая необходимые функциональные особенности среды, было принято решение использовать ИСРПО Borland Delphi седьмой версии, рабочее пространство которой представлено на рисунке 3.

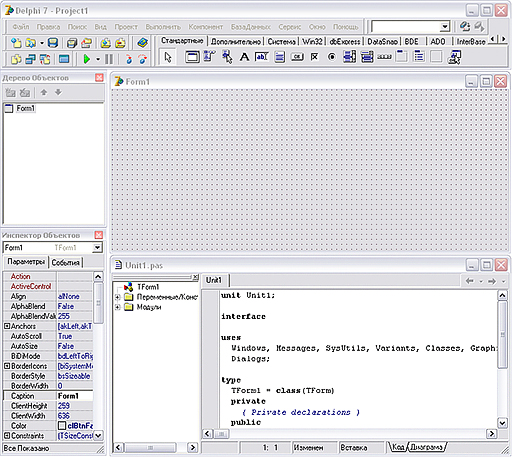


Рисунок 3 – Рабочее пространство ИСРПО Borland Delphi 7

Borland Delphi 7 включает в себя текстовый редактор, компилятор, отладчик, справочную систему, средства для визуального проектирования форм программы (в т.ч. VCL). Текстовый редактор использует подсветку синтаксиса языка для наглядного восприятия кода и всплывающее меню подсказок при написании и обращении к различным функциям и классам. Компилятор и отладчик, встроенные в Delphi 7, обеспечивают высокую производительность программы, легкость и быстрое время проверки готового программного блока. Средства для визуального проектирования форм позволяют достаточно быстро и качественно реализовать интуитивно понятный графический интерфейс пользователя.

Помимо инструментального программного обеспечения, еще одной составляющей средств разработки является программные средства тестирования и документирования ПО. Для данных целей выберем свободно распространяемый (бесплатный) мощный офисный пакет OpenOffice.org 3.2, разработчиком которой является Oracle Corporation (ранее Sun Microsystems) совместно с сообществом программистов OpenOffice.org. В его состав входит текстовой процессор Writer, табличный процессор Calc и другие приложения, которые не уступают по функциональности приложениям его платного аналога – офисного пакета Microsoft Office. OpenOffice.org Calc позволяет анализировать вводимые данные, производить сложные вычисления, строить различные диаграммы, с его помощью будут подготавливаться тестовые наборы для проверки создаваемой программы. OpenOffice.org Writer будет использоваться как средство для создания отчетов по тестированию, хранения и печати различных промежуточных данных по разработке, и формирования документации конечного пользователя ПО.

Косвенной, но, тем не менее, также важной составляющей средств разработки является операционная система (ОС), без которой невозможна работа всех вышеописанных частей средств разработки. С учетом выбранного языка программирования, ИСРПО и средств документирования в данном дипломном проекте будет использоваться 32-разрядная операционная система Microsoft Windows XP для платформы X86. Несмотря на конкретно выбранную ОС, создаваемая программа будет работать и на других 32/64-разрядных операционных системах семейства Windows, таких как Windows 95/98/Me/NT/2000/XP, включая последние версии Windows Vista/Seven и на ее серверных вариантах.

Таким образом, был осуществлен выбор средств разработки: проведен анализ и выбор языка программирования и интегрированной среды разработки программного обеспечения, а также выбрана операционная система и средства для создания тестов и документации к программному изделию. Применение в средствах разработки выбранного языка программирования и свободного программного обеспечение позволят сократить денежные затраты на обучение языку и приобретение ПО.

**Рефлексия работы над кейсом**

* Почему выбрана данная проблема в области создания ПО?
* Почему выбрана данная тема и схема исследования?
* Какие этапы вы можете выделить в работе по теме занятия?
* Каков алгоритм исследования?
* Какие особенности, удачи и трудности хотелось бы отметить в работе над проектом?

**Задание**

При оформлении отчета по практическому занятию № 1 «Анализ существующих альтернативных программных продуктов. Исследование рынка программного обеспечения» следует составить свой кейс по тематике занятия. Все ссылки на использованные источники следует оформить в соответствии с требованиями ГОСТ.