**Модернизация ИС**

Главной причиной модернизации информационной системы является расхождение между требованиями к информационной системе со стороны предприятия и ее действительными характеристиками. Такое расхождение имеет тенденцию к нарастанию со временем. Относительно небольшое расхождение позволяет говорить о необходимости модернизации ИС, сильное –о необходимости модернизации информационной системы. Основными причинами, также приводящими к модернизации информационных систем, являются :

 моральное устаревание информационной системы (информационных технологий, пользовательских и программных интерфейсов, используемых в составе ИС);

физическое устаревание информационной системы (износ ее аппаратных компонентов);

причины организационного характера (связанные с окружением информационной системы, бизнес-процессами предприятия, пользователями системы).

Моральное устаревание ИС в основном вызвано появлением:

 более эффективных информационных технологий (ИТ);

новых способов организации пользовательского интерфейса;

 новых решений в области архитектуры информационной системы;

вычислительных устройств с более высокой производительностью;

новых носителей информации (более дешевых, с большим быстродействием, позволяющих хранить больше информации).

Физическое устаревание ИС в основном вызвано:

физическим износом используемого аппаратного обеспечения (снижением надежности, увеличением количества сбоев);

ухудшением характеристик производительности аппаратного обеспечения (снижение быстродействия из-за больших объемов накопленной информации).

Основной причиной организационного характера для модернизации информационной системы является развитие предприятия (как штатным образом, так и в результате модернизации бизнес-процессов), совершенствование его бизнес-процессов. Все это требует обновления и развития информационной поддержки бизнес-процессов предприятия. Кроме того, постоянно растет квалификация персонала, что позволяет внедрять более сложные информационные технологии и проводить информатизацию все новых сфер деятельности. Со временем ситуация с расхождением между требованиями к ИС и ее характеристиками становится критической (в настоящем или будущем и требуется серьезное вмешательство в информационную систему. Часто причиной модернизации ИС является модернизации бизнес-процессов. И наоборот, модернизации ИС часто приводит к РБП. В любом случае, модернизации ИС требует коррекции бизнес-процессов предприятия.

Системное приложение «Книги» имеет минусы: малую функциональность, недочеты в плане оптимизации и д.р.

В будущем, необходимо обновление приложение с полной поддержкой пользовательской корзины и ее основного содержания в виде отдельной страницы, нового функционала и обновлением старого.

Требуется поддержка мониторов с высоким разрешением (адаптивность приложения под особенности пользователя), а также адаптация под новые системы.

Кроме-того нужна поддержка базы данных, в дальнейшем с оптимизированием добавления данных в нее.

Добавление функций для корректной работы подсчета скидок и общей стоимости приобретаемого товара.  
Защита приложения от некорректного пользовательского использования.

При программировании вопрос об эффективности работы программ встает только при нехватке каких-либо ресурсов. Например, памяти при работе с большими массивами данных или времени при сложных громоздких расчетах или интенсивной работе с данными. При технических характеристиках современных компьютеров многие программы выполняются практически мгновенно. Тем не менее задачи написания эффективных программ встречаются на олимпиадах. Кроме этого четвёртую задачу ЕГЭ второй части с развёрнутым ответом также предлагается решить с учётом эффективности выполнения. Поэтому вопрос о написании эффективных программах актуален и для школьников.

**Способы повышения эффективности программ**

1. Способы уменьшения времени выполнения
   1. Время выполнения программы в первую очередь зависит от используемых в ней методов – нужно использовать более **эффективные алгоритмы.**
   2. В том случае, когда в программе выполняется большое количество арифметических вычислений, для повышения скорости работы программы необходимо правильно программировать арифметические выражения. Различные арифметические операции различаются по быстродействию. Самыми быстрыми являются операции **сложения** и **вычитания**. Более медленным является **умножение**, затем идёт **деление**. Поэтому операция x/a выполняется медленнее, чем x\*b, где b=1/a, операция 2\*x выполняется медленнее, чем x+x.
   3. Программируя арифметические выражения, следует выбирать такую форму их записи, чтобы количество «медленных» операций было сведено к минимуму. Например, пусть необходимо вычислить: **ax4+bx3+cx2+dx+e**, где содержится 10 умножений («медленных» операций) и 4 сложения («быстрых» операций). Это же самое выражение можно записать в виде: (((**ax+b)x+c)x+d)x+e**. Такая форма записи называется схемой Горнера. В этом выражении 4 умножения и 4 сложения. Общее количество операций сократилось почти в два раза, соответственно уменьшится и время вычисления выражения. Подобные оптимизации являются алгоритмическими.
   4. Увеличивают время выполнения программы циклические фрагменты с большим количеством повторений. Поэтому по возможности необходимо **минимизировать тело цикла.** Выражения, фрагменты которых никак не зависят от управляющей переменной цикла, называются инвариантными фрагментами. При написании циклов рекомендуется выносить их из тела цикла.
   5. При использовании вложенных циклов следует иметь в виду, что затраты процессорного времени на обработку такой конструкции могут зависеть от порядка следования вложенных циклов. При программировании вложенных циклов по возможности следует **делать цикл с наименьшим числом повторений самым внешним, а цикл с наибольшим числом повторений — самым внутренним**.
2. Способы экономии памяти(представлены ознакомительно и в работе не анализировались)
   1. Следует выбирать алгоритмы обработки, не требующие дублирования исходных данных структурных типов. Например, не использовать дополнительные массивы.
   2. По возможности использовать динамическую память. При необходимости выделять память, а потом освобождать.
   3. При передаче структурных данных в подпрограмму по значению, копии этих данных размещаются в стеке. Избежать копирования можно, если передавать данные не по значению, а как неизменяемые (описанные const). В последнем случае в стеке размещается только адрес данных.
3. **Вычисление точного времени выполнения конкретного блока программы**

Для экспериментальной проверки найденных способов повышения эффективности программ, то есть их оптимизации, необходимо было найти способ вычисления времени выполнения отдельных фрагментов программы. Трудность заключалась в том, что в среде программирования PascalABC.NET нет процедуры вычисления времени. Поэтому в работе использовалась небольшая программа (см. Приложение 1), найденная в Интернете [4]. В программу вставлялись контрольные точки, например, фиксировалось время начала выполнения цикла и время его окончания с помощью этой программы, а затем определялось время выполнения нужного фрагмента программы.

Другой проблемой стало ничтожно малое время выполнения составленных программ из-за небольшого количества повторений тела цикла, которое просто не фиксировалось программой вычисления времени. Для того, чтобы увеличить счетчик повторений цикла и менять его оперативно для анализа различных входных данных пришлось задавать элементы массива случайным образом, а количество повторений цикла вынести в раздел описания констант.

Своевременное выполнение обновлений поможет избежать следующих негативных последствий:

* развитие информационной системы под требования бизнеса только за счет собственных разработок;
* увеличение количества собственных разработок приводит только к установке нот и обновлений, устраняющих ошибки и необходимых для выполнения требований изменения законодательства. Со временем установка таких обновления становится все более трудоемкой;
* увеличение стоимости поддержки информационных систем;
* завершение поддержки производителей устаревших баз данных и операционных систем;
* снижение общего уровня безопасности системы.

**Обновление**Процесс обновления систем заключается в последовательном выполнении следующих этапов:

* *Обследование*. На этом этапе определяется уровень обновлений системы, подключение дополнительных функциональных возможностей. Проводится сбор информации об объеме внедрения бизнес-процессов, операций по ним и определяется объем тестирования. Также осуществляется анализ объема, модифицированного ПО и собственных разработок.
* *Подготовка плана перехода.* Производится подготовка тестовой системы (копия продуктивной), ее обновление, анализ и корректировка затронутого модифицированного ПО; тестирование работы системы, регистрация и решение проблем; создание перечня мероприятий для перехода. Затем проводится повторное разворачивание тестовой системы, ее обновление, тестирование и применение плана мероприятий. Производится планирование сроков этапов перехода, оценка рисков и возможность дополнительных мероприятий по их снижению. В итоге определяется период неработоспособности продуктивной системы, разрабатывается и утверждается документ «План перехода»
* *Выполнение плана перехода*. Заключается в последовательном выполнении мероприятий, описанных в документе "План перехода".
* *Поддержка пользователей.*После переноса обновлений в продуктивную систему заказчика осуществляется оперативная поддержка пользователей и решение оставшихся проблем.

**Миграция**Процесс миграции систем включает в себя следующие этапы:

* *Обследование*. На этом этапе определяется перечень мероприятий.
* *Подготовка плана миграции.* Производится применение перечня мероприятий, подготовка стенда, проверяется работоспособность системы. Затем проводится тестовая миграция, в ходе которой уточняется, обновляется перечень мероприятий, а также определяется их длительность. На основе результатов этого процесса производится планирование сроков этапов миграции, оценка рисков и возможность дополнительных мероприятий по их снижению. В итоге определяется период неработоспособности продуктивной системы, разрабатывается и утверждается документ «План перехода»
* *Выполнение плана миграции*. Заключается в последовательном выполнении мероприятий, описанных в документе.
* *Поддержка пользователей.*После миграции осуществляется оперативная поддержка пользователей и решение оставшихся проблем.
* В будущем, необходимо обновление приложение с полной поддержкой пользовательской корзины и ее основного содержания.  
  Требуется поддержка мониторов с высоких разрешением (адаптивность приложения под особенности пользователя).  
  Добавление функций для корректной работы подсчета скидок и общей стоимости приобретаемого товара.  
  Защита приложения от некорректного пользовательского использования.