Частное учреждение образования

«Колледж бизнеса и права»

УТВЕРЖДАЮ

Ведущий

методист колледжа

\_\_\_\_\_\_\_\_\_\_\_ Е.В. Паскал

«\_\_» \_\_\_\_\_\_\_\_\_\_\_\_\_\_2021

|  |  |
| --- | --- |
| Специальность: 2-40 01 01 «Программное обеспечение информационных технологий» | Дисциплина: «Технология разработки программного обеспечения» |

# Лабораторная работа №19

**Инструкционно-технологическая карта**

Тема: Создание прототипа

Цель работы: Получить навыки разработки и отладки программных модулей, научиться анализировать полученные результаты с целью модернизации кода

Время выполнения: 2 часа

**Контрольныевопросы**

1. Назовите критерии оценки программного приложения с точки зрения декомпозиции его на модули
2. Опишите процесс кодирования модулей программной системы

**Теоретические сведения для выполнения работы**

Приступая к разработке программного приложения, следует иметь в виду, что большие по объёму кода системы автоматизации разрабатывают по частям, которые называются *программными модулями*. Такой метод разработки программ называют *модульным программированием*. Программный модуль − это любой фрагмент описания процесса, оформляемый как самостоятельный программный продукт, пригодный для использования в описаниях процесса. Это означает, что каждый программный модуль программируется, компилируется и отлаживается отдельно от других модулей программы, и тем самым, физически разделен с другими модулями программы. Более того, каждый разработанный программный модуль может включаться в состав разных программ, если выполнены условия его использования, декларированные в документации по этому модулю. Таким образом, программный модуль может рассматриваться и как средство борьбы со сложностью программ, и как средство борьбы с дублированием в программировании (т.е. как средство накопления и многократного использования программистских знаний).

Программы разбиваются на модули для того, чтобы:

* упростить их разработку и реализацию;
* облегчить чтение программ;
* упростить их настройку и модификацию;
* облегчить работу с данными, имеющими сложную структуру;
* избежать чрезмерной детализации алгоритмов;
* обеспечить более выгодное размещение программ в памяти ЭВМ.

Не всякий программный модуль способствует упрощению программы. Выделить хороший с этой точки зрения модуль является серьезной творческой задачей. Для оценки приемлемости выделенного модуля используются некоторые критерии, например:

* хороший модуль снаружи проще, чем внутри;
* хороший модуль проще использовать, чем построить.

Можно использовать более конструктивные характеристики, например:

* размер модуля;
* прочность модуля;
* сцепление с другими модулями;
* рутинность модуля (независимость от предыстории обращений к нему).

*Размер* *модуля* измеряется числом содержащихся в нем операторов или строк. Модуль не должен быть слишком маленьким или слишком большим. Маленькие модули приводят к громоздкой модульной структуре программы и могут не окупать накладных расходов, связанных с их оформлением. Большие модули неудобны для изучения и изменений, они могут существенно увеличить суммарное время повторных трансляций программы при отладке программы. Обычно рекомендуются программные модули размером от нескольких десятков до нескольких сотен операторов.

*Прочность (связность)* *модуля* − это мера его внутренних связей. Чем выше прочность модуля, тем больше связей он может спрятать от внешней по отношению к нему части программы и, следовательно, тем больший вклад в упрощение программы он может внести.

*Сцепление модуля* − это мера его зависимости по данным от других модулей. Характеризуется способом передачи данных. Чем слабее сцепление модуля с другими модулями, тем сильнее его независимость от других модулей.

*Рутинность модуля* − это его независимость от предыстории обращений к нему. Модуль будем называть рутинным, если результат (эффект) обращения к нему зависит только от значений его параметров (и не зависит от предыстории обращений к нему). Модуль будем называть зависящим от предыстории, если результат (эффект) обращения к нему зависит от внутреннего состояния этого модуля, изменяемого в результате предыдущих обращений к нему. Майерс не рекомендует использовать зависящие от предыстории (непредсказуемые) модули, так как они провоцируют появление в программах хитрых (неуловимых) ошибок. Однако такая рекомендация является неконструктивной, так как во многих случаях именно зависящий от предыстории модуль является лучшей реализаций информационно прочного модуля. Поэтому более приемлема следующая (более осторожная) рекомендация:

* всегда следует использовать рутинный модуль, если это не приводит к плохим (не рекомендуемым) сцеплениям модулей;
* зависящие от предыстории модули следует использовать только в случае, когда это необходимо для обеспечения параметрического сцепления;
* в спецификации зависящего от предыстории модуля должна быть четко сформулирована эта зависимость таким образом, чтобы было возможно прогнозировать поведение (эффект выполнения) данного модуля при разных последующих обращениях к нему.

В связи с последней рекомендацией может быть полезным определение внешнего представления (ориентированного на информирование человека) состояний зависящего от предыстории модуля. В этом случае эффект выполнения каждой функции (операции), реализуемой этим модулем, следует описывать в терминах этого внешнего представления, что существенно упростит прогнозирование поведения данного модуля.

**Порядок выполнения работы**

1. Изучите теоретические сведения.
2. Разработайте программный модуль, состоящий из нескольких взаимосвязанных форм по теме индивидуального задания и в соответствии с проектом интерфейса, созданным при выполнении лабораторной работы №15 (графическая заставка для инициализации работы; форма для регистрации пользователя и ввода исходных данных, форма для заполнения и редактирования базы данных проекта, форма отчетности).

**Домашнее задание**

Оформите отчет и ответьте на контрольные вопросы.

Преподаватель \_\_\_\_\_\_\_\_ А.С. Комаровский

Рассмотрено на заседании цикловой

комиссии ПОИТ №10

Протокол № от «\_\_\_» \_\_\_\_\_\_\_\_\_\_ 2021

Председатель ЦК \_\_\_\_\_\_\_\_ В.Ю. Михалевич