**Лабораторная работа No4.**

**Тема:** Изучение методологий структурного проектирования ПО.

**Цель работы:**

1. Изучить основные подходы, используемые при проектировании программного продукта структурным способом.

2. Научиться использовать нисходящее и восходящее проектирование для разбиения исходной задачи на подзадачи с их последующей детализацией на следующих уровнях.

**Ответы на контрольные вопросы:**

1. Проектирование любого программного продукта основано не только на выборе модели ЖЦ ПО и разделении работ по его созданию на стадиях проектирования, но и на выборе подхода к проектированию. Наиболее распространенные методологии структурного подхода базируются на ряде общих принципов. В качестве двух базовых принципов используются следующие:

Принцип "разделяй и властвуй" – принцип решения сложных проблем путем их разбиения на множество меньших независимых задач, легких для понимания и решения;

Принцип иерархического упорядочивания – принцип организации составных частей проблемы в иерархические древовидные структуры с добавлением новых деталей на каждом уровне.

К достоинствам данного подхода относятся, прежде всего:

–  возможность проведения глубокого анализа бизнес-процессов, выявления узких мест: комплексное применение позволяет выявить все возможные рассогласования и неточности;

–  применение универсальных графических языков моделирования IDEF0, IDEF3 и DFD обеспечивает логическую целостность и полноту описания, необходимую для достижения точных и непротиворечивых результатов;

–  проверенность временем и широкое распространение среди аналитиков и разработчиков.

В качестве недостатков можно выделить такие:

–  низкая наглядность для неподготовленных пользователей модели: при увеличении количества уровней представления, анализа и модификации моделей становится затруднительными;

–  сложность восприятия иерархически упорядоченной информации;

–  необходимость следования жесткой (не всегда необходимой) структуре.

1. Основными принципами проектирования являются: объективность, прогрессивность, перспективность, комплексность, нормативность, последовательность, экономичность.

Принцип объективности выражает необходимость всестороннего, конкретного и объективного обоснования целесообразности каждого проектного решения и доказательства: необходимости создания, возможности осуществления и эффективности применения предлагаемого проектного решения.

**Возможность**осуществления проектных решений доказывают путем изучения реальных условий, наличия объективных данных, которые позволяют превратить возможность в действительность.

**Эффективность** применения новых проектных решений доказывают, основываясь на определении показателей технической, экономической и социальной эффективности при сравнении с действующими аналогичными решениями.

**Принцип прогрессивности** конкретное выражение принципа прогрессивности заключается в том, что к моменту ввода в эксплуатацию запроектированного производственного объекта его проектные решения должны иметь более высокий уровень по сравнению с передовыми действующими отечественными и зарубежными аналогичными объектами.

**Принцип перспективности** производственный объект рассчитывают на эксплуатацию в течение нескольких десятков лет. За пределами обозримого будущего его перспективы не однозначны, поэтому принцип перспективности требует учета в проекте возможностей в отдаленном будущем дальнейшего развития и совершенствования проектных решений.

Комплексный характер каждого проектного решения является его объективным свойством, сущность которого заключается в том, что любое проектное решение одновременно является простым и сложным. Простым, по отношению к проектному решению, в состав которого оно входит, и сложным, по отношению к проектным решениям, которые входят в его состав.

1. Структурное проектирование понимается как методология построения алгоритмов, программ и систем, в том числе информационных, в основе которой лежит выявление структуры задачи, определение составляющих компонент и выделение связей между ними. Процесс разделения сложных задач (объектов, систем) на относительно независимые друг от друга подзадачи (части, подсистемы) называется декомпозицией.

Выделяют следующие типичные методы структурного проектирования:

* - нисходящее проектирование;
* - модульное программирование.

Нисходящее проектирование (или проектирование«сверху вниз»*)* заключается в применении процедуры пошаговой детализации: сложная задача разбивается на более простые, которые, в свою очередь, разбиваются далее до тех пор, пока вся задача не окажется реализованной.

Модульное программирование связано с созданием и использованием специальных программных единиц (программных) модулей*.* Каждый модуль содержит некоторое количество подпрограмм, которые используют одни и те же глобальные данные (рис. 10.11). Также в модули объединяют подпрограммы, относящиеся к одной функциональной группе, например, математические, подпрограммы обработки графики. Подпрограммы модулей могут быть использованы как в основной программе, так и в других модулях.

