

## Билет 2

### 1. Основные подходы к разработке ПО САПР.

Сейчас в программной инженерии есть *два основных подхода к разработке ПО* ИС, принципиальная разница между которыми обусловлена различными способами декомпозиции систем: структурный и объектно-ориентированный подход.

Структурный подход связан с функциональной декомпозицией системы, выделением независимых компонентов, таких как: подсистемы, функции, процедуры обработки данных, образующие иерархию. При этом должна быть обеспечена целостность при ограничении функциональности, обозримости и модифицируемости отдельных компонентов. Результатом структурного подхода является разработка модульной архитектуры ПО, отделение данных от программ их обработки, что обеспечивает независимость представления логической и физической структуры данных. В качестве инструментальных средств проектирования структурный подход использует: диаграммы потоков данных – DFD (Data Flow Diagram); методы структурного анализа и проектирования – SADT (Structured Analysis and Design Technique; диаграммы «сущность–связь» для реляционных БД – ERD (Entity Relationship Diagrams).

Объектно-ориентированный подход использует объектную декомпозицию ИС и ПО. «Объекты» – предметы, процессы или явления, обладающие набором уникальных свойств (данных) и методов их обработки, инициируемых в момент возникновения предопределенных событий. Объекты объединяют в себе как данные, так и программный код. Объекты разделяются на классы. Внутри класса объектов создаются представители, которые наследуют свойства и методы класса. При этом для любого объекта возможно изменение наследуемых свойств и методов класса, а также передача методов обработки (программного кода) другим объектам. Объектный подход позволяет изолировать объект от многообразия всей системы, закончить разработку объекта, тиражировать программный код. Объектно-ориентированный подход вызвал к появлению огромное число инструментальных средств, объектно-ориентированных языков программирования.

### 2. Принципы построения и функционирования ЭВМ. Принцип программного управления.

*Архитектура вычислительной системы* - общая логическая организация цифровой вычислительной системы, определяющая процесс обработки данных в конкретной вычислительной системе и включающая методы кодирования данных, состав, назначение, принципы взаимодействия технических средств и программного обеспечения.

Все универсальные вычислительные машины, в том числе и персональные компьютеры, имеют структуру, показанную на рис. 1, где обозначено:

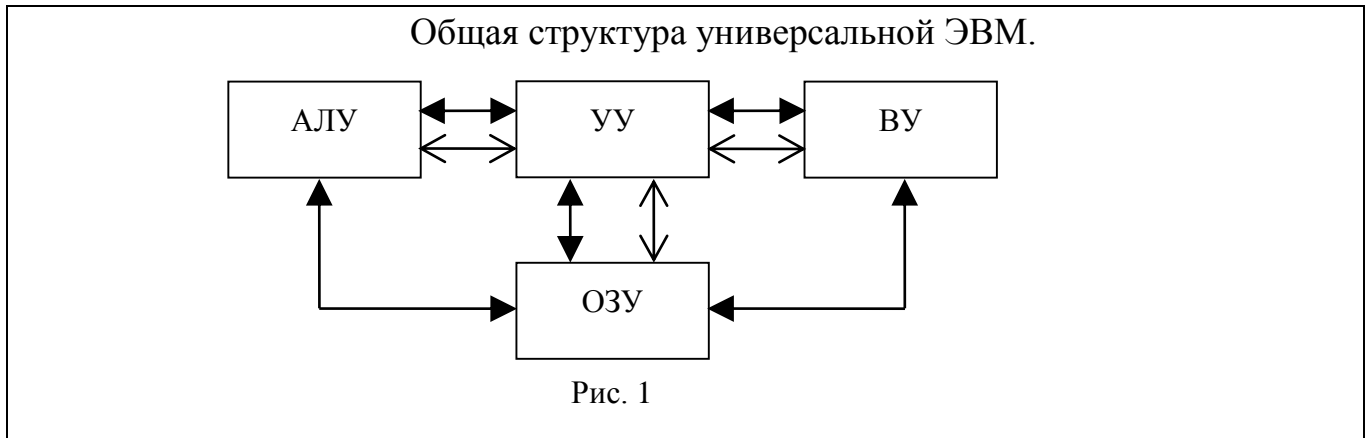
АЛУ - арифметическо-логическое устройство;

УУ - устройство управления;

ВУ - внешние устройства;

ОЗУ - оперативное запоминающее устройство.

Линии со стрелками  $\longleftrightarrow$  и  $\rightleftarrows$  означают информационные и управляющие связи, соответственно.



Впервые такую структуру вычислительных машин предложил Джон фон Нейман в 1945 г., поэтому ЭВМ со структурой на рис. 1 называют машинами фон Неймана. Чтобы компьютер был универсальным и эффективным устройством для обработки информации, он должен иметь следующие основные устройства:

- *арифметическо-логическое устройство*, выполняющее арифметические и логические операции;
- *устройство управления*, предназначенное для организации процесса выполнения программ;
- *оперативное запоминающее устройство* (оперативная память) для хранения программ и данных;
- *внешние устройства для ввода/вывода информации*.

Построение вычислительных машин основано на трех принципах:

- принцип цифрового представления данных (чисел, команд, обозначение операций, букв, слов и т.д.). Единицами данных в ЭВМ являются бит, байт, слово и т.п.;
- принцип адресности данных, согласно которого все данные и любые объекты программы хранятся в ячейках памяти, имеющих адрес;
- принцип программного управления (Ч. Беббидж, 1834 г.), сущность которого состоит в том, что управление вычислительным процессом осуществляется с помощью программы, находящейся в памяти ЭВМ.

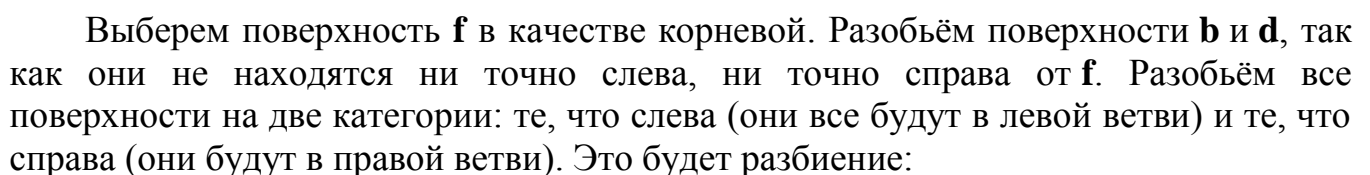
### ***Принцип программного управления***

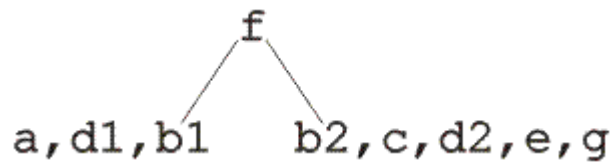
В XIX веке английским математиком и инженером Чарльзом Бэббиджем был разработан проект вычислительной машины, которая предназначалась для автоматического проведения длинных цепочек вычислений. Конструкция его аналитической машины включала 50 тысяч деталей. В созданной Бэббиджем

Главной особенностью конструкции этой машины является **программный принцип** работы. Принцип программы, хранимой в памяти компьютера, считается важнейшей идеей современной компьютерной архитектуры. Суть идеи заключается в следующем:

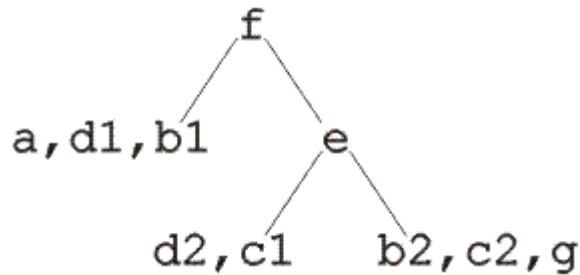
- ### 3. Показать пример построения BSP-дерева в алгоритмах удаления невидимых граней.

Для того чтобы объяснить принцип работы BSP дерева лучше начать с примера. Рассмотрим рисунок комнаты.

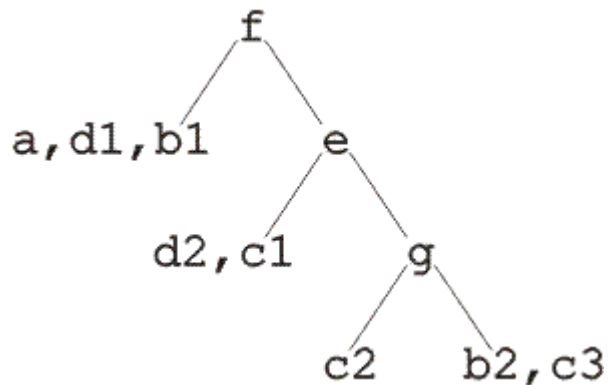




Мы можем завершить формирование левой ветви. Вершины **a,d1,b1** никогда не будут зрительно перекрывать друг друга. С другой стороны, то есть справа, поверхность **e** может иногда закрывать **d2**. Так что мы выбираем **e** и делим **c** помощью поверхности **e** как **c** с помощью поверхности **f**. Это заставляет нас разбить **c**. Мы разбили все поверхности так:



Затем, разобьём по поверхности **g**, разбивая опять **c**, и все получившиеся разбиения будут представлять собой листья.



Будем разбивать до тех пор, пока не останется по одной поверхности в вершине:

