

# Глава 3.

## «Структура пользовательского интерфейса»



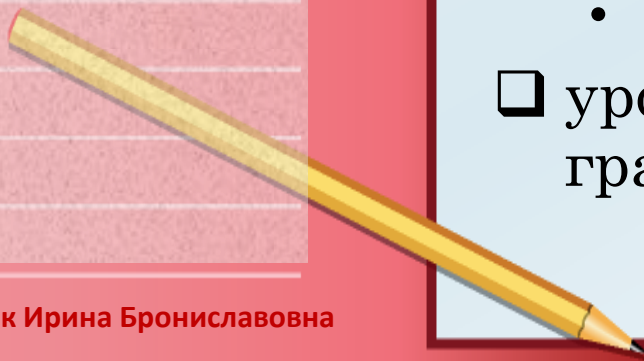
## Тема 8. Категории пользователей



1. Диалоговый компонент прикладной системы.
2. Схема алгоритма диалога с программным приложением.
3. Типы диалога.
4. Сценарий взаимодействия.
5. Структуры диалога.



## **Факторы, влияющие на отношение пользователя к системе**

Рассмотрим факторы,  
влияющие на отношение  
пользователя к  
интерактивной системе и  
на методы его работы с ней.



- 
- ☐ **человеческие потребности**  
(потребность быть понятым  
партнером по диалогу);
  - ☐ **навыки пользователя:**
    - моторные навыки (работа с  
клавиатурой и мышью),
    - лингвистические навыки (в  
общении и в решении задач);
  - ☐ **свойства личности:**
    - творческие способности,
    - подверженность ошибкам,
    - способность к обучению,
    - терпеливость,
    - устойчивость к стрессу и т.д.;
  - ☐ **уровень компьютерной  
грамотности.**
- 

# Классификации пользователей интерактивных систем

*По уровню компьютерной грамотности*



Бухгалтер



Чайник

Различают:

- ☐ программирующих,
- ☐ не программирующих пользователей.

Среди последних, в свою очередь, выделяют три категории:

- ☐ подготовленные пользователи, решающие творческие задачи, аналитики и исследователи (то есть пользователи, последовательность действий которых сложно формализовать);
- ☐ подготовленные пользователи, выполняющие рутинные операции, операторы (то есть пользователи, последовательность действий которых является достаточно устойчивой);
- ☐ случайные (или “наивные”) пользователи, обладающие минимальным уровнем компьютерной грамотности.

# Классификации пользователей интерактивных систем


## Подготовка в прикладной области

*Подготовка в прикладной области*  
пользователя влияет на  
использование языка (например,  
профессиональной  
терминологии) и применяемые  
методы решения задач.


**Теза́урус** (от греч. *θησαυρός* — сокровище) в современной лингвистике — особая разновидность словарей общей или специальной лексики, в которых указаны семантические отношения (синонимы, антонимы, паронимы, гипонимы, гиперонимы и т. п.) между лексическими единицами.

**Дескрипторы** — слова или словосочетания, соответствующие основным понятиям предметной области системы и включённые в её тезаурус.

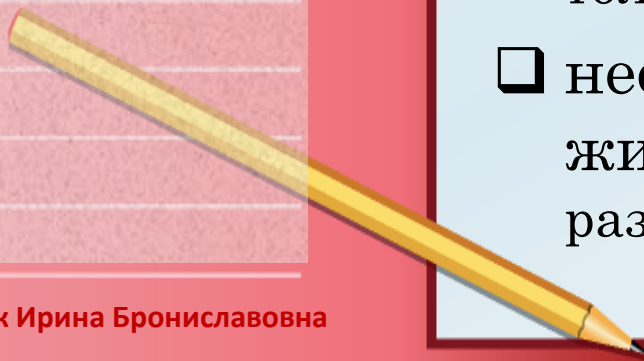




# Причины пользования системой




*Причина пользования системой  
может быть:*

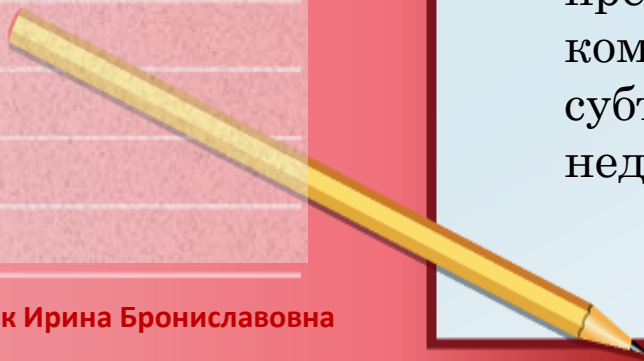
- ☐ обязательной – как неотъемлемая часть работы;
  - ☐ необязательной – как дополнительная составляющая профессиональной деятельности;
  - ☐ обязательной с точки зрения личных потребностей (например, когда определенную информацию можно получить только с помощью компьютера);
  - ☐ необязательной в частной жизни (например, в качестве развлечения).
- 



## Отношение пользователей к системе



*Отношение к системе* и ожидания от работы с ней определяются уровнем компьютерной грамотности и причиной ее использования, оно может быть:

- ☐ **нейтральным:** пользователь воспринимает компьютер только как рабочий инструмент;
  - ☐ **положительным:** пользователь охотно использует компьютер, так как надеется получить от него пользу (или удовольствие);
  - ☐ **негативным:** пользователь предпочел бы не пользоваться компьютером (возможно, из-за субъективной нелюбви или недоверия к технике).
- 

# Цели пользователя

*Ограничения по времени*  
независимо от  
характеристик системы  
(в частности, ее  
быстродействия), в этом  
случае пользователь может  
быть вынужден  
приспосабливаться к  
ограничениям по времени,  
исходящим от задачи или  
контекста работы.

*Целями пользователя*  
могут быть:

- ☐ решить определенную задачу с помощью компьютера (в том числе и развлекательного характера);
- ☐ научиться работать с системой.



# Типы операторской деятельности

В зависимости от уровня иерархии управления, на котором находится человек-оператор, и типа системы управления (или обработки информации) можно выделить следующие типы операторской деятельности:



- оператор – технолог, непосредственно включенный в технологический процесс и выполняющий стандартные процедуры управления;
- оператор – манипулятор, реализующий процедуры управления, состоящие из механических воздействий;
- оператор – наблюдатель, следящий за состоянием процесса и его отклонениями в системах реального времени;
- оператор – исследователь, действия которого не регламентированы имеющимися процедурами и базируются на понятийном мышлении (программист);
- оператор – руководитель, выполняющий функции организационного и директивного характера;
- оператор – проектировщик, включенный в процесс машинного проектирования в составе САПР.



# Деятельность человека- оператора


Существует ряд общих положений, свойственных всем (или большинству) типам операторов (например, общая модель мышления и принятия решений человеком, его психофизиологические характеристики и др.).

В процессе решения задач выделяется 5 этапов, характерных практически для всех категорий операторов:

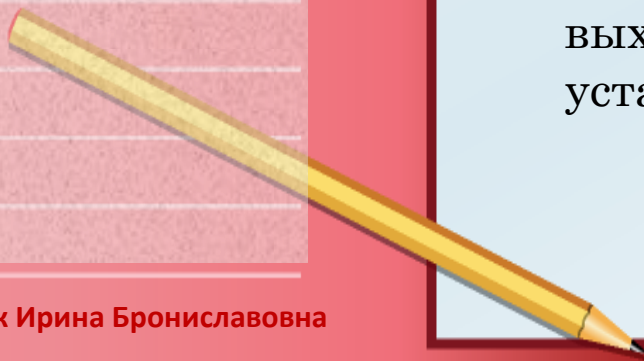
- ❑ **обнаружение** – восприятие оператором группы признаков, инициирующих решение задачи;
- ❑ **классификация** – распознавание ситуации и определение состояния контролируемого процесса и технической системы;
- ❑ **планирование** – принятие решения и разработка последовательности действий, которые предположительно приведут к решению задачи;
- ❑ **исполнение** – реализация планов с учетом имеющихся ресурсов;
- ❑ **отслеживание** – проверка исполнения действий и их эффективности.




# Задачи, решаемые оператором




Задачи, решаемые оператором, в свою очередь, делятся на 3 категории:


- ☐ **регулирование процесса**, предусматривающее незначительное динамическое выравнивание и корректировку отдельных параметров процесса;
  - ☐ **изменение режима**, ведущее к принципиальному изменению процесса и предусматривающее переходные процессы, большое количество операций и сложные алгоритмы действий оператора;
  - ☐ **деятельность в аномальных ситуациях**, проявляющихся в выходе параметров процесса за установленные границы.
- 



# Типы поведения человека- оператора



В зависимости от сложности и формализованности задач, выделяется 3 уровня поведения оператора:

- ❑ **поведение, основанное на навыках** – действия оператора рефлексорны и ориентированы, от него не требуется вникать в суть происходящих процессов, он должен лишь реагировать заранее определенным способом на возникновение некоторых ситуаций;
  - ❑ **поведение, основанное на правилах** – деятельность оператора связана с использованием правил управления, сведенных в инструкции по эксплуатации системы;
  - ❑ **поведение, основанное на знаниях** – действия оператора требуют понимания протекающих процессов и обработки информации, явно не содержащейся в текущих данных или в заранее подготовленных инструкциях.
- 



# Типы моделей, используемых оператором

В процессе решения задач оператор может использовать 5 типов моделей, формализующих как его действия, так и описание контролируемых процессов:

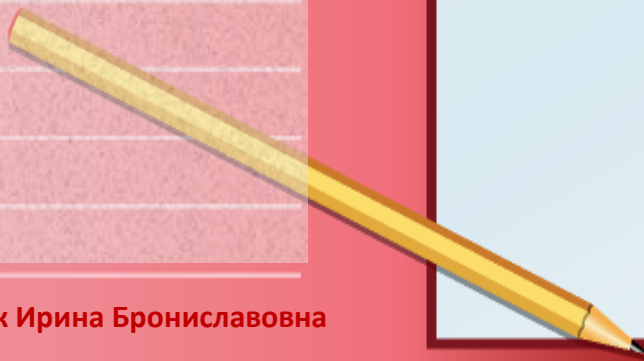
- ☐ **физическую**, содержащую математическое описание процессов, протекающих в системе;
- ☐ **функциональную**, описывающую основные подсистемы, оборудование, его функции и взаимосвязи;
- ☐ **экономическую**, связывающую величины стоимости основных операций управления;
- ☐ **процедуральную**, описывающую правила управления установкой;
- ☐ **когнитивную**, отражающую интуитивную модель процесса, формируемую оператором.







## Глава 3.

### «Структура пользовательского интерфейса»




## Тема 9. Структуры и типы диалога.

1. Классификации пользователей интерактивных систем.
  2. Типы операторской деятельности.
  3. Этапы деятельности человека-оператора.
  4. Типы поведения человека-оператора.
  5. Типы моделей, используемых оператором.
- 




## **Функциональные компоненты диалоговой системы**



В прикладной диалоговой системе можно выделить два функциональных компонента:

- ☐ собственно прикладную систему, с которой работает пользователь;
- ☐ диалоговый компонент, управляющий диалогом между пользователем и прикладной системой.



# Примеры прикладных систем

Это все системы, с которыми работают пользователи ЭВМ в сфере своих профессиональных интересов



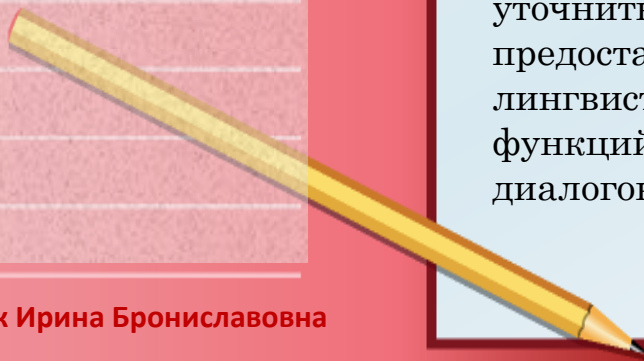
Примеры прикладных систем:

- ☐ различного рода автоматизированные системы управления (АСУ),
- ☐ системы автоматизированного проектирования (САПР),
- ☐ пакеты прикладных программ (ППП),
- ☐ и др.

Эти системы характеризуются тем, что диалог ведется в терминах проблемной области и на выходе пользователь получает конечный результат, являющийся целью его общения с программным приложением.

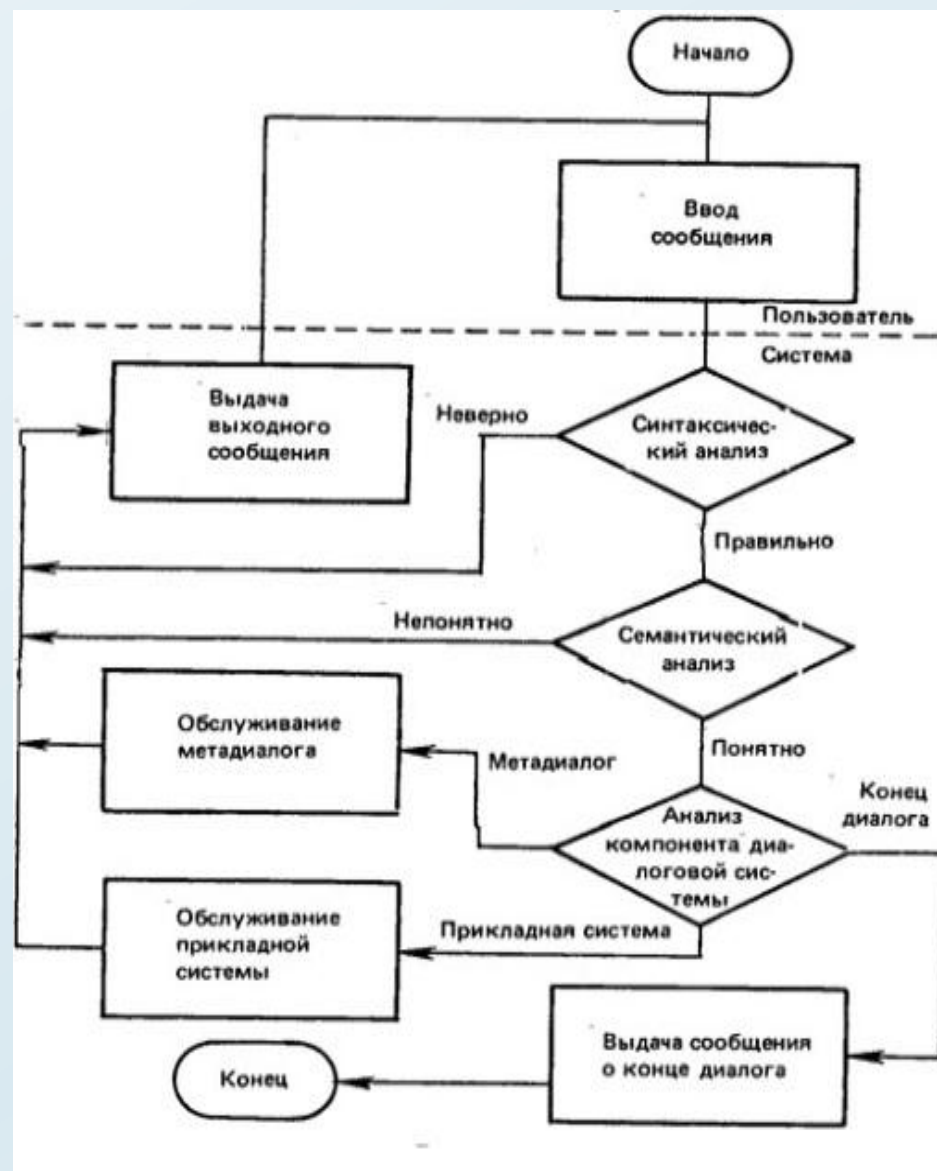
Некоторые системы могут работать не только в диалоговом, но и в пакетном режиме. Семантика конечного результата при этом не зависит от используемого режима работы.

Диалоговый компонент является надстройкой над прикладной системой и помогает пользователю уточнить постановку проблемной задачи, предоставляя для этого соответствующие лингвистические формы и набор сервисных функций. Диалог в этом случае ведется в терминах диалоговой системы и называется *метадиалогом*.



# Обобщенная схема алгоритма диалога

С учетом наличия двух компонентов в диалоговой системе обобщенную схему алгоритма диалога с программным приложением можно представить.






# Обобщенная схема алгоритма диалога

При этом однократное прохождение по одной из ветвей схемы алгоритма от блока “Начало” до блока “Конец” или до “Возврат к началу” схемы алгоритма будем называть *шагом диалога* или *транзакцией*.


Восприятие системой входного сообщения включает 2 этапа:

- ❑ **синтаксический анализ** – проверка соответствия формы, в которой введена информация, разрешенным на данном шаге, диалога шаблонам или правилам ввода-вывода;
- ❑ **семантический анализ** – выявление смысла входного сообщения, определение ближайших задач системы и анализ возможности их выполнения при данном состоянии процедурной части.


Рассмотренная последовательность начинается с вывода системой выходного сообщения, т.е. инициатором взаимодействия является система.



## Распределение инициативы в диалоге




Формы распределения  
инициативы в диалоге:

- ☐ инициатива,  
принадлежащая  
пользователю (когда  
пользователь сам управляет работой  
системы посредством команд и  
запросов),
  - ☐ смешанная инициатива,  
(предполагающая периодическое  
перераспределение инициативы с  
помощью управляющих сигналов, с  
помощью которых пользователь  
имеет возможность прервать  
исполнение как процедурной, так и  
диалоговой части программы и  
перевести ЭВМ в состояние  
ожидания своей команды).
- 



# Типы диалогов




По характеру информации, содержащейся во входных и выходных сообщениях, различаются 2 типа диалога:

- ☐ *управляющий*, когда в результате транзакции определяются функции системы и осуществляется управление ее работой (для этой цели используются также и управляющие-сигналы);
- ☐ *информационный*, в процессе которого в систему вводятся данные, требуемые для решения конкретной задачи.



## Классификация диалоговых систем




В зависимости от числа пользователей диалоговые системы могут быть:

- ☐ ***однопользовательскими***  
(рассчитанными на одного пользователя)?
- ☐ ***многопользовательскими***  
(с коллективным доступом пользователей к ресурсам системы).





# Формы диалога



Для реализации транзакций диалога используются следующие формы:

- ☐ *синтаксически ограниченная* (меню, запросы с ответом по шаблону, запросы с ответом ДА/НЕТ);
- ☐ *директивная (команды);*
- ☐ *фразовая* (взаимодействие на ограниченном естественном языке);
- ☐ *объектно-ориентированная.*

# Типы диалога

Для более быстрого освоения диалоговой системы пользователю необходимо знать основные этапы диалога и формы его реализации.

Наиболее известны в человеко-машинной среде следующие типы диалога:

- ☐ меню;
- ☐ вопросы, требующие ответа ДА/НЕТ;
- ☐ шаблон;
- ☐ простой запрос;
- ☐ команда;
- ☐ взаимодействие на естественном языке.

# Сценарий взаимодействия

Взаимодействие человека-оператора с ЭВМ обычно подчиняется определенному сценарию диалога.

Его проектируют, исходя из задач, стоящих перед АСОИиУ.

АСОИиУ (автоматизированные системы обработки информации и управления)

Сценарий диалога представляет собой алгоритм взаимодействия оператора с программной системой.

Сценарий диалога представляет собой детальное описание диалогового взаимодействия, выраженное в виде структуры диалога, а также его информационной и операционной моделей.

Формально сценарий определяется следующим образом:


$$D = \langle S, A, C, R, G, I, O \rangle,$$

где

- $S$  – множество состояний;
- $A$  – множество операций;
- $C = Q \cup F$  – множество условий ( $Q$  – множество входных условий,  $F$  – множество программных условий);
- $R$  – множество выходных сообщений;
- $G: S \times C \rightarrow S$  – граф диалога;
- $I: \{S \rightarrow R; S \times C \rightarrow R\}$  – информационная модель диалога;
- $O: \{S \rightarrow A; S \times C \rightarrow A\}$  – операционная модель диалога.



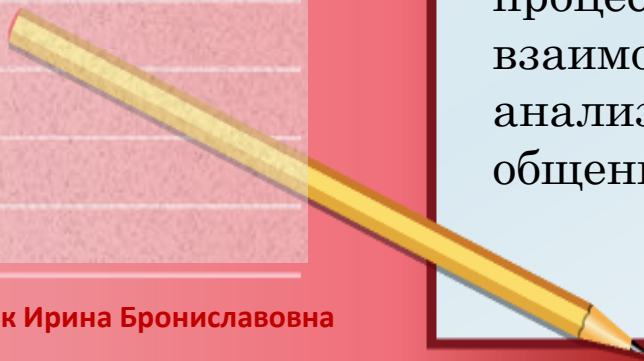
## Сценарий взаимодействия



Ведение диалога согласно сценарию называется *интерпретацией диалога*.

Графическое изображение сценария напоминает блок-схему программы на алгоритмическом языке, состоящую из элементов ветвления, ввода вывода сообщений и исполнения операций. Элементарная конструкция такой блок-схемы называется *узлом сценария*.

При проектировании взаимодействия пользователя с программной системой *сценарий диалога* является ключевым понятием. Он позволяет формализовать процесс разработки диалогового взаимодействия, а также проводить анализ и оптимизацию процедур общения пользователя с системой.





# Структура диалога

Рассмотрим основную составную часть сценария – структуру диалога, наиболее наглядно отображающую процесс взаимодействия.

**Структура диалога** – это связанная совокупность состояний диалога достижимых в процессе общения пользователя с диалоговой системой.

Состояния диалога складываются из трех компонент:

- ☐ формы диалога;
- ☐ совокупности функций системы, предоставляемой пользователю;
- ☐ предыстории диалога.

Основное графическое представление структуры диалога – диаграмма состояний – *ориентированный взвешенный граф*.

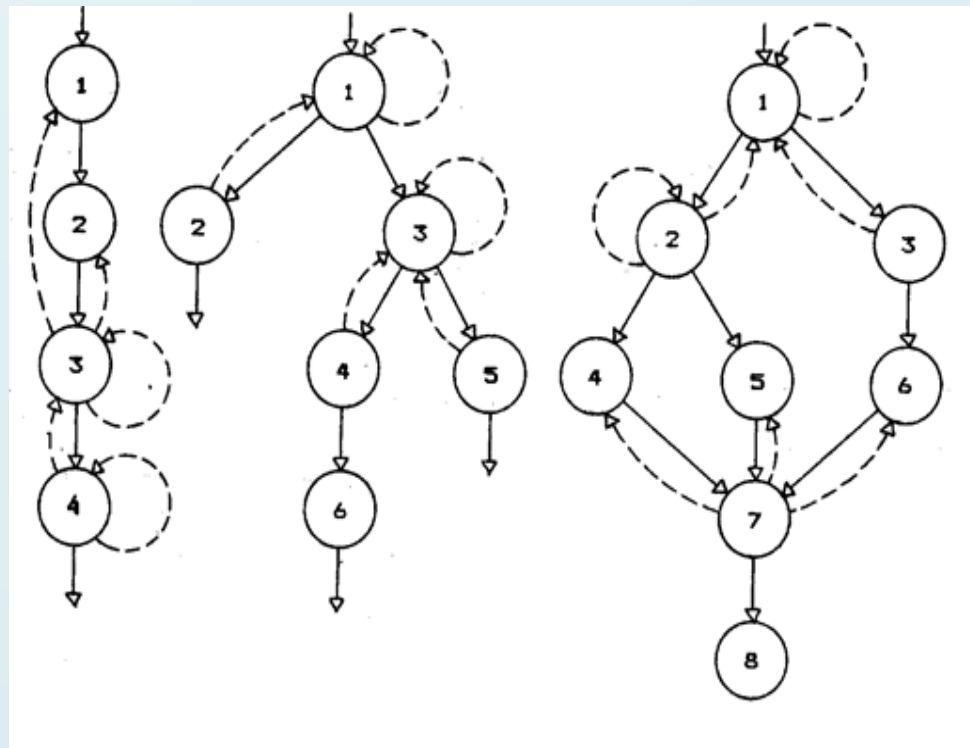
Каждая вершина графа соответствует *определенному состоянию диалога*, а дуга определяет *смену состояний*.

Веса вершин и дуг характеризуют *содержание диалога*: входные и выходные сообщения, исполняемые программные операции и условия.

Переход от одной вершины графа диалога к другой соответствует одной *транзакции диалога*.

Произвольная последовательность переходов (путь по графу) образует *сеанс общения* пользователя с системой.

# Структура диалога



Можно выделить три структуры диалога:


- ☐ линейную,
- ☐ древовидную
- ☐ сетевую



# I. Диалог в форме “Меню”

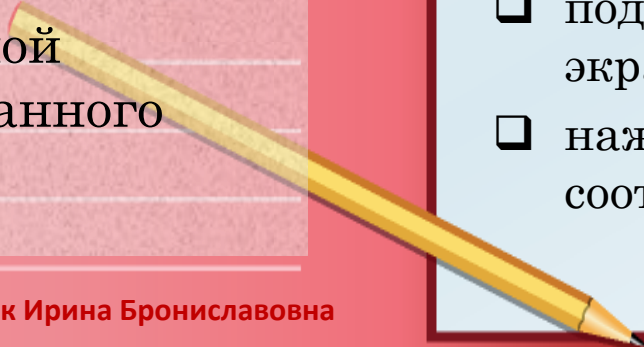

*Меню* – наиболее распространенный тип диалога.

Возможно несколько форм машинной реализации данного типа диалога.



Во всех случаях в качестве входного сообщения на экране дисплея высвечивается подмножество (как правило, нумерованное) функций системы, реализация которых возможна в текущем состоянии диалога. Такое входное сообщение будем называть *меню*.

Пользователь может выбрать требуемую функцию одним из следующих основных способов:

- ☐ набором требуемой директивы или ее аббревиатуры;
  - ☐ набором номера выбранной директивы;
  - ☐ подведением курсора в строку экрана с требуемой директивой;
  - ☐ нажатием фотоселектора в соответствующей строке экрана.
- 
- 

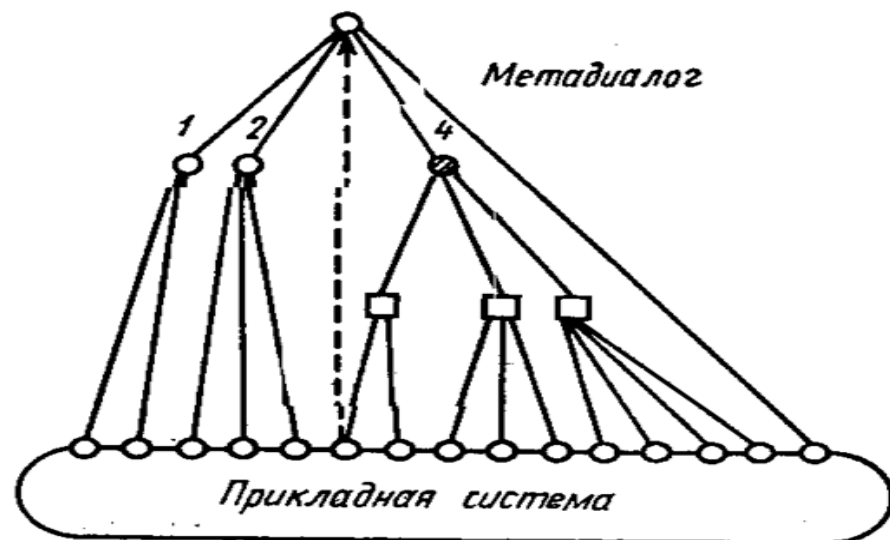


# 1. Диалог в форме “Меню” (способ 1)

Корень дерева и исходящие из него дуги соответствуют шагу диалога.

После выполнения некоторых шагов диалога система высветит меню, обозначенных прямоугольниками.

Движение вниз по дереву графа меню в направлении прикладной системы называется “**погружением**”



Графически диалог можно представить в виде графа.

Узлы графа соответствуют подмножеству функций системы, а дуги, направленные вниз, - возможным альтернативам пользователя.

При этом число дуг равно числу предлагаемых системой функций.

Дуга вместе с вершиной, из которой она исходит, соответствует шагу диалога.


Верхний узел (корень) дерева соответствует начальному сообщению системы.





# 1. Диалог в форме “Меню” (способ 2)

Графически второй способ соответствует наличию в каждом узле ориентированной к вершине дерева дуги графа. Одна из таких дуг изображена штриховой линией.



Для выполнения функции, соответствующей соседним ветвям дерева графа, при работе с прикладной системой используется механизм “всплытия”.

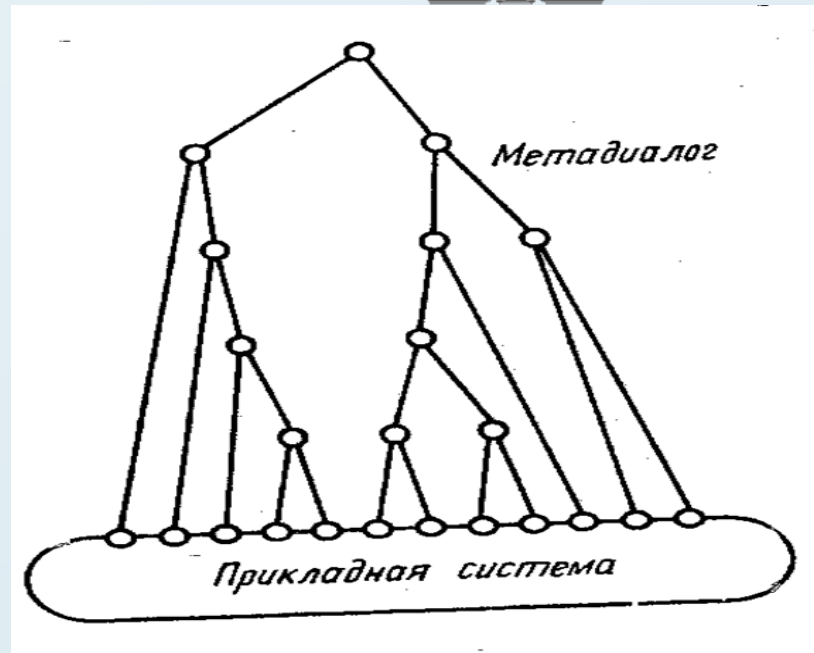
Его можно реализовать двумя способами:

- ☐ последовательным движением вверх по дугам графа диалога до узла, непосредственно содержащего заданную функцию или содержащего ее в узлах, в которые из него можно перейти, используя “погружение”;
- ☐ непосредственным переходом в корень дерева графа диалога с последующим погружением.

## II. Диалог в форме “Вопросов, требующих ответа ДА/НЕТ”

Диалог в форме вопросов, требующих ответа ДА/НЕТ – тип диалога, являющийся частным случаем диалога типа “меню”, когда в шаге диалог; в качестве альтернатив пользователю предлагаются два ответа: ДА и НЕТ.

Основным достоинством данной формы диалога является простота выполнения каждого отдельного шага диалога.



Это граф бинарного дерева, из каждого узла которого выходит вниз (к прикладной системе) по две ветви.

“Погружение” и “всплытие” в этом типе диалога аналогичны меню.

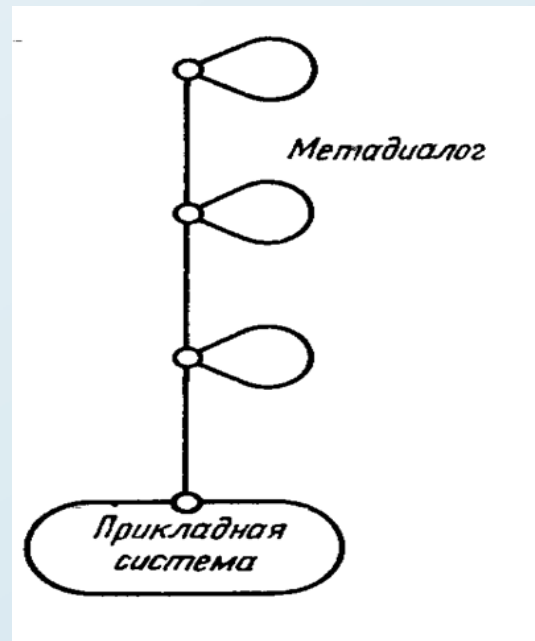
В данном типе диалога для работы с прикладной системой приходится выполнять, как правило, больше шагов по сравнению с диалогом типа “меню”.

### III. Диалог в форме “Шаблона”

*Шаблон*— это инициированный системой диалог, на каждом шаге которого система воспринимает только синтаксически ограниченное входное сообщение пользователя в соответствии с заданным форматом.

Графически диалог типа “шаблон” можно интерпретировать с помощью графа

Данный тип диалога называют также *параметрической системой*.



Как и в меню, диалог начинает система. Возможные варианты ответа пользователя ограничиваются форматами, предъявляемыми ему на дисплее. Поэтому гибкость пользования системой с диалогом такого типа относительно невысока, однако достаточно низка ее операционная сложность.



### III. Диалог в форме “Шаблона”

Из каждой вершины графа выходит по две дуги.

Дуга, направляющаяся к следующему узлу, соответствует семантически правильному сообщению пользователя, а дуга, замыкающаяся на одном узле, — семантически неправильному сообщению пользователя.

При неверном сообщении система выдаст предупреждение об ошибке и останется в начале текущего шага диалога.

Шаблон обычно используется для ввода данных, значения которых или “прозрачны” (например, текущая дата), или являются профессиональными терминами и поэтому они известны пользователю.

Однако формы подачи этих данных не определяются их значениями.



## IV. Диалог в форме “Простого запроса”

*Простой запрос* – это тип диалога, инициируемый системой и состоящий из одного шага, где пользователю предоставляется возможность ввести более одного сообщения по заранее определенному системой единому формату.

Графическая  
интерпретация  
простого запроса



Простой запрос является частным случаем шаблона.

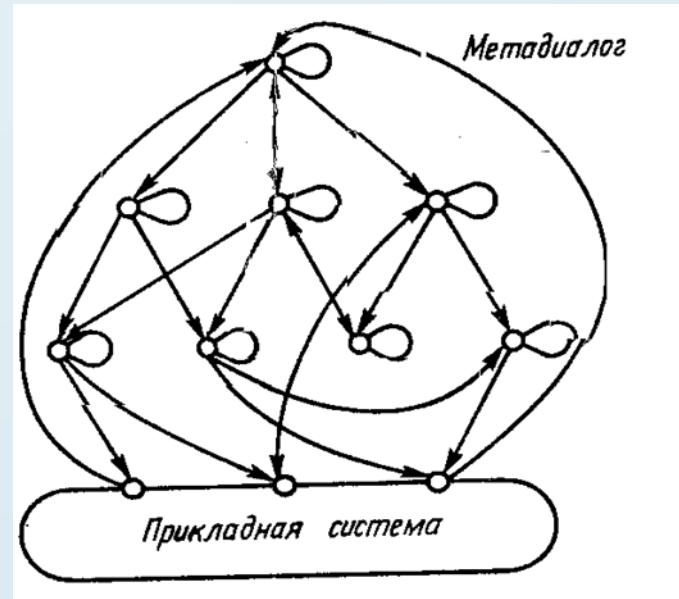
Обычно используется в различного рода автоматизированных системах при сборе данных.

# Диалог в форме “Команда”

*Команда* – это инициированный пользователем тип диалога, при котором выполняется одна из допустимых на данном шаге диалога директива (команда) пользователя.

Характерным является наличие дуг, замыкающихся на каждом узле графа. Такие дуги соответствуют недопустимым директивам на данном шаге диалога.

Графически данный тип запроса может быть представлен с помощью ориентированного графа

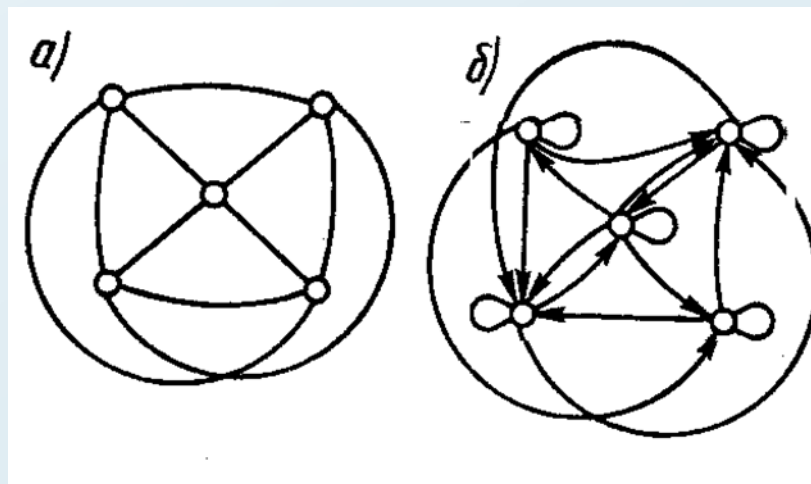


Набор допустимых команд, как правило, отсутствует на экране, однако с помощью специальных директив его можно вывести для ознакомления.

В случае задания недопустимой директивы выдается предупредительное сообщение об ошибке, и система остается в начале текущего шага диалога.

## V. Диалог в форме “Взаимодействие на естественном и квазиестественном языках”

*Взаимодействием на естественном языке будем называть тип диалога, инициированный пользователем и приводящий к решению поставленной им задачи, когда общение со стороны пользователя ведется на естественном языке.*



Графическая интерпретация взаимодействия на языках:

- а) естественном,
- б) квазиестественном.

Этот тип диалога налагает наименьшие ограничения на форму ведения общения стороны пользователя, которому предоставляется возможность свободно выбирать формулировку задачи, используя естественный язык.



## V. Диалог в форме “Взаимодействие на естественном и квазиестественном языках”

Научные и технические трудности создания системы, взаимодействующей с пользователем на естественном языке, приводят к реализации *квазиестественного* (похожего на естественный) или проблемно-ориентированного (для общения в одной узкой проблемной области) языка.

Особенностью графа диалога на естественном языке является наличие парных противоположно направленных дуг графа между любыми парами узлов (на рисунке каждая пара противоположно направленных дуг обозначена одной неориентированной дугой).

Граф диалога на квазиестественном языке является ориентированным, характерная его особенность – наличие в каждом узле дуг, замыкающихся на этом же узле, что соответствует недопустимым директивам на данном шаге диалога.

В отличие от графа диалога на естественном языке на квазиестественном не все пары узлов соединены противоположно направленными дугами.



# Вывод

## Анализ типов диалога


Анализ описанных типов диалога показывает, что применение только одного из них (за исключением последнего) не позволяет создать достаточно гибкую и эффективную систему.

Рассмотрим ограничения на структуру диалога, накладываемые разными его типами:

- ❑ *вопросы, требующие ответа ДА/НЕТ* – мала эффективность диалога, велика избыточность графа диалога при усложнении темы общения, чрезмерное их использование приводит к переутомлению пользователя;
- ❑ *шаблон* – эффективен только в случае ввода фрагмента табличных данных, при ограниченном (от двух до четырех) выборе возможных альтернатив и малом числе шагов диалога;
- ❑ *команда* – мала эффективность при значительном числе (не меньше пяти) шагов диалога и альтернативных команд на каждом его шаге. Командная форма диалога удобна для опытных пользователей, команды дают возможность сразу специфицировать необходимую функцию вместо того, чтобы отвечать на длинную серию запросов.



# Вывод:



Использование в диалоговой системе нескольких типов диалога повышает гибкость системы и снижает ее операционную сложность.

Работая с диалоговой системой, пользователь должен быть готов к тому, что ему придется встретиться с несколькими типами диалога.