

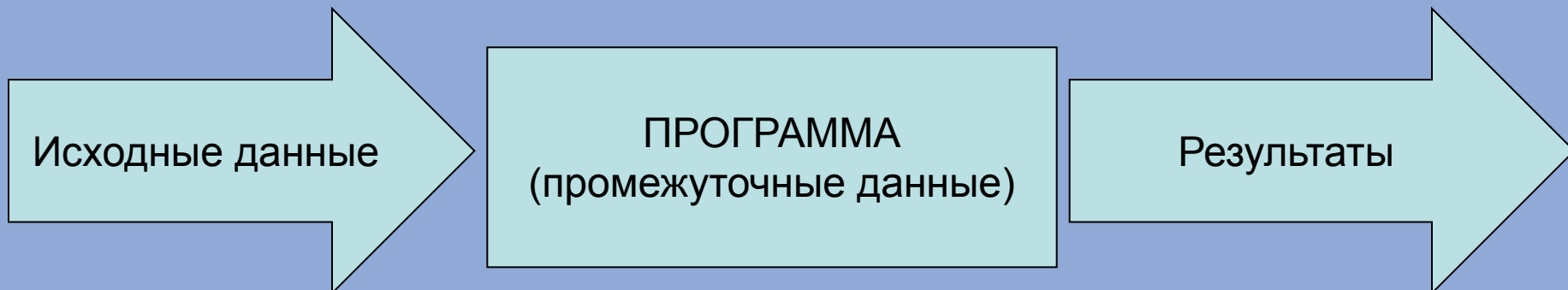
Данные.

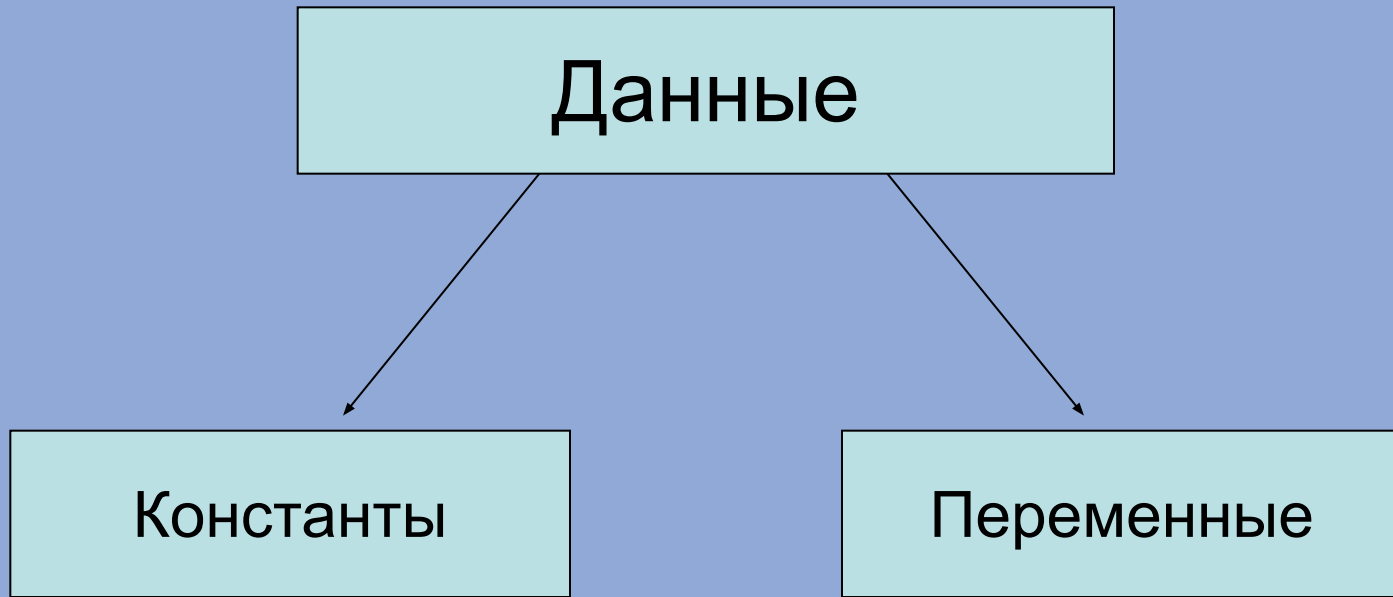
Тип данных

Данные - это любая информация, представленная в формализованном виде и пригодная для обработки алгоритмом.

По отношению к программе данные делятся на:

- исходные;
- результаты (окончательные данные);
- промежуточные. Которые получают в процессе вычислений.





Константы – это данные, значения которых не меняются в процессе выполнения алгоритма.

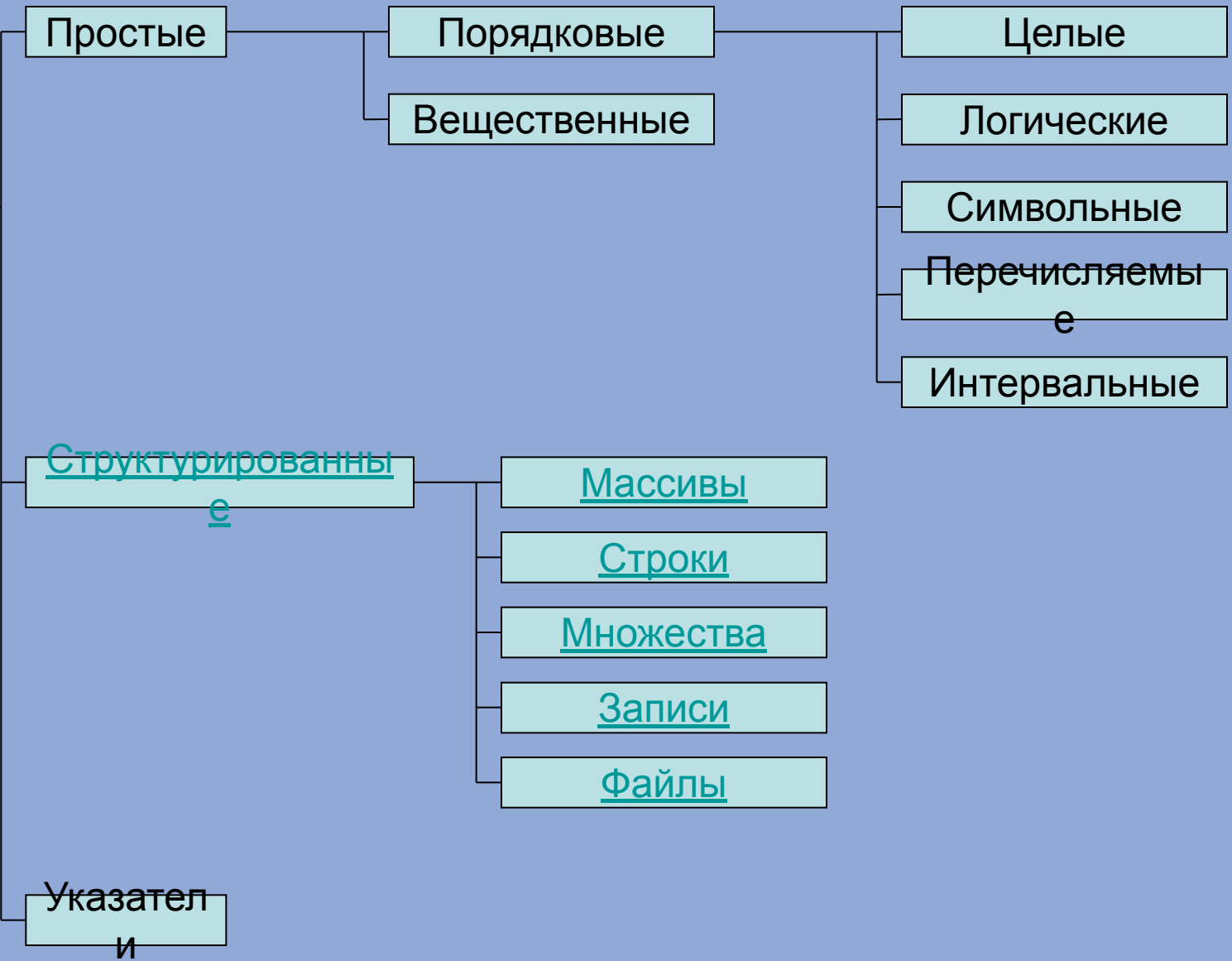
Переменные – это такие данные, значения которых могут изменяться в процессе выполнения алгоритма.

Каждая переменная и константа должна иметь свое уникальное *ИМЯ* и характеризуются *типом данных*. Имена переменных и констант задаются идентификаторами.

Идентификатор представляет собой последовательность букв и цифр, начинающаяся с буквы.

Тип данных определяет множество значений каждого элемента данного типа, а также множество операций, которые выполняются над этими элементами

ТИПЫ



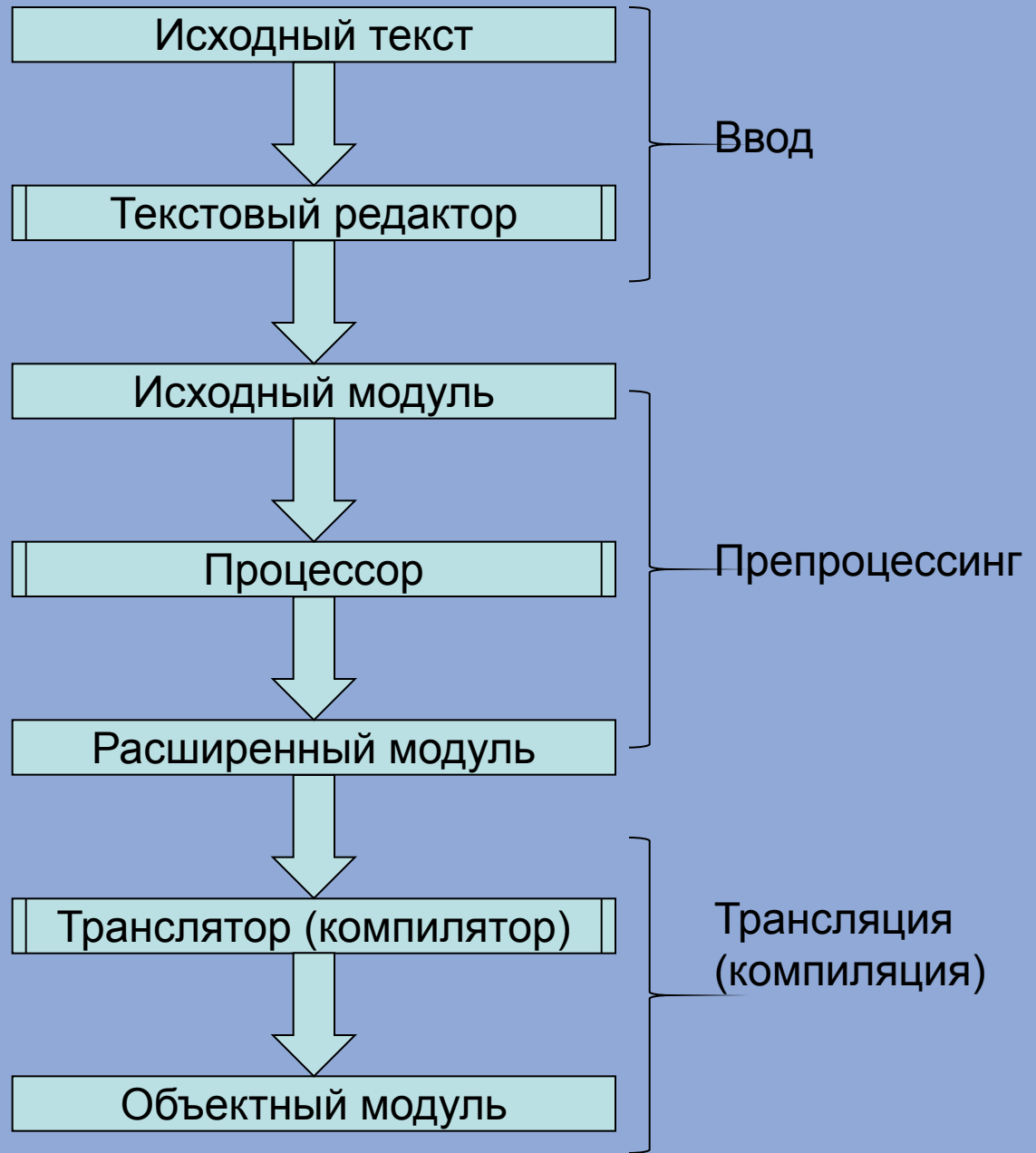
Тип	Значения	Операции	Внутреннее представление
Целый	Целые положительные и отрицательные числа в некотором диапазоне.	Арифметические операции с целыми числами: +, -, x, целое деление и остаток от деления. Операции отношений (<, >, = и др.)	Формат с фиксированной точкой
Вещественный	Любые (целые и дробные) числа в некотором диапазоне. Примеры: 2,5, -0,01,45,0,3,6x10 ⁹	Арифметические операции: +, -, x, /. Операции отношений	Формат с плавающей точкой
Логический	True(истина), False(ложь)	Логические операции: И (and), ИЛИ(or), НЕ (not). Операции отношений	1 бит: 1 – true; 0 – false;
Символьный	Любые символы компьютерного алфавита. Примеры: 'a', '5', '+', '\$'	Операции отношений	Коды таблицы символьной кодировки. 1 символ – 1 байт

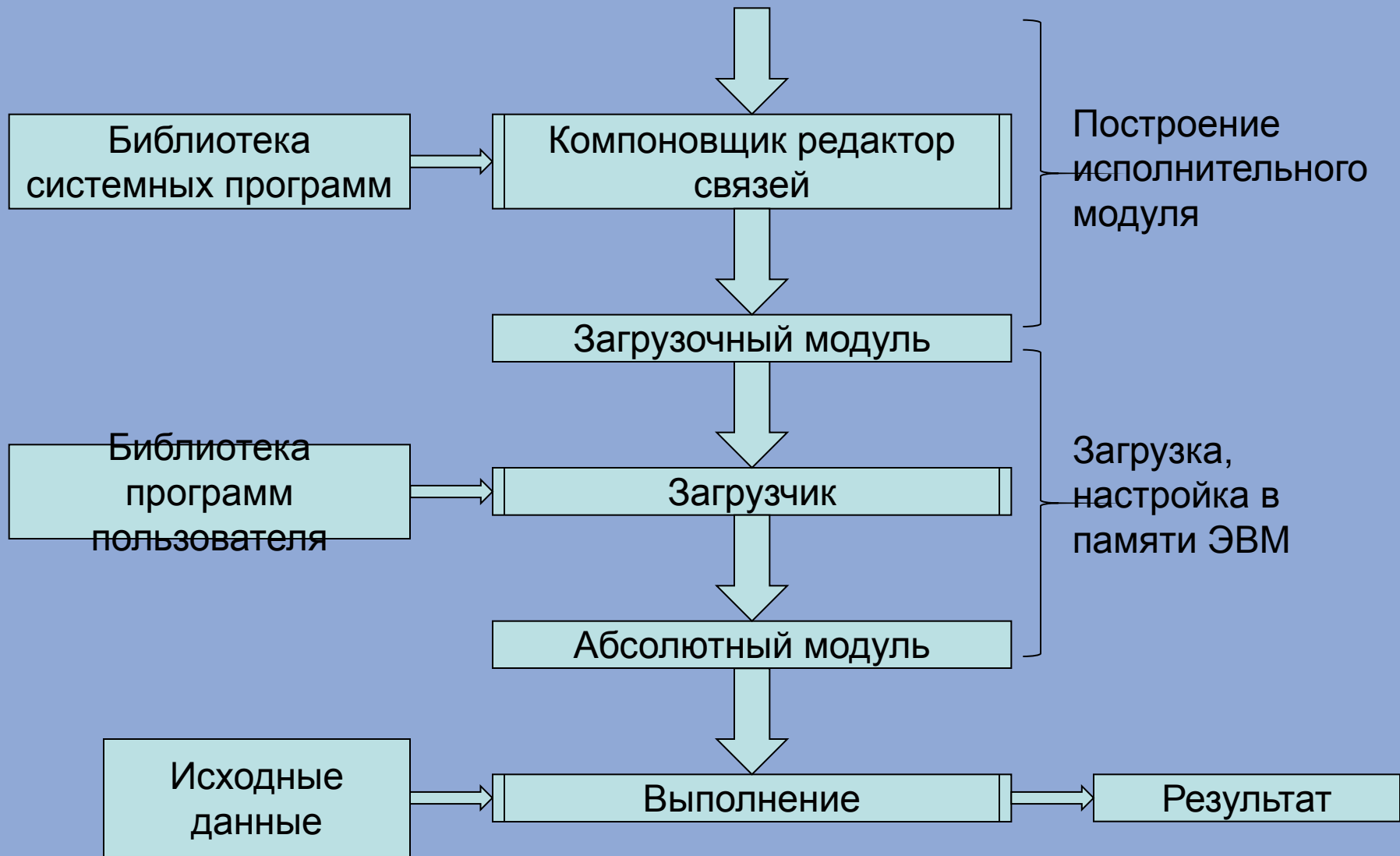
Структурированный тип – это набор однотипных или разнотипных данных, с которыми алгоритм работает как с одной именованной переменной

Системы программирования

Система программирования – интегрированная среда, предназначенная для автоматизации процесса программирования, которая содержит трансляторы алгоритмических языков высокого уровня и инструментальную программную оболочку, включающую редактор текстов программ, отладчики и другие вспомогательные программы

Схема разработки прикладных программ в среде системы программирования





Ввод. Программа на исходном языке (исходный модуль) готовится с помощью текстовых редакторов и в виде текстового файла или раздела библиотеки поступает на вход транслятора.

Препроцессинг – необязательная фаза, состоящая в анализе исходного текста, извлечения из него директив процессора и их выполнения.

Трансляция. Трансляция исходной программы есть процедура преобразования исходного модуля в промежуточную, так называемую объективную форму. Трансляция в общем случае включает в себя препроцессинг (предобработку) и компиляцию.

Компиляция

В общем случае многоступенчатый процесс, включающий следующие фазы:

- **синтаксический анализ** – проверка правильности конструкций, использованных программистом при подготовке текста;
- **семантический анализ** – выявление несоответствий типов и структур переменных, функций и процедур;
- **генерация объектного кода** – завершающая фаза трансляции.

Объектный модуль представляет собой текст программы на машинном языке, включающий машинные инструкции, словари, служебную информацию. Он не работоспособен.

Построение исполнительного модуля. Построение загрузочного модуля осуществляется специальными программными средствами – редактором связей, построителем задач, компоновщиком, основной функцией которых является объединение объектных и загрузочных модулей в единый загрузочный модуль с последующей записью в библиотеку или файл. Полученный модуль в дальнейшем может использоваться для сборки других программ и т.д., что создает возможность наращивания программного обеспечения.

Загрузка программы. Загрузочный модуль после сборки либо помещается в качестве раздела в пользовательскую библиотеку программ, либо в качестве последовательного файла на накопителе на магнитном диске

Современные системы программирования позволяют удобно переходить от одного этапа к другому. Это осуществляется в рамках так называемой интегрированной системы программирования, которая содержит в себе текстовый редактор, компилятор, компоновщик, встроенный отладчик и, в зависимости от системы или её версии, предоставляет программисту дополнительные удобства для написания и отладки программ.

**Массивы – упорядоченная
последовательность
переменных одного и того же
типа, имеющая общее имя.**

Запись (record) – агрегат данных, составляющий элемент базы данных (файла), содержащий разнотипную информацию, описывающие некоторый объект (сущность, экземпляр, аспект).

Файл – именованный
организованный набор
данных определенного типа и
названия, находящийся под
управлением операционной
системы

Строка – это последовательность
СИМВОЛОВ.

1 символ = 1 байт памяти (код
ASCII).

Количество символов = длиной.

**Множество - совокупность
однотипных элементов,
рассматриваемых как единое
целое.**

Этапы решения задач на ЭВМ

1. *Постановка задачи* – на этом этапе определяется при каких условиях задача может быть решена, а также исходные данные и результат полученный при решении
2. *Построение математической модели* – при этом текстовая запись задачи должна быть переведена на язык математических терминов

3. *Выбор метода решения задачи* – зависит от модели которая была получена на предыдущем этапе
4. *Построение алгоритма решения задачи.* Результатом данного этапа является блок-схема. Если для решения задачи можно предложить несколько алгоритмов, то выбирается один эффективный.
5. *Построение программы.*
6. *Отладка программы* – исправление синтаксических ошибок

7. *Тестирование программы.* Для тестирования берут один вариант исходных данных и просчитывают в ручную, потом подставляют в программу, при совпадении результатов можно сделать вывод о правильности, если обнаружена ошибка, то её необходимо искать начиная с первого этапа
8. *Эксплуатация программы*
9. *Анализ работы программы*

Спецификация

спецификация - это точное описание функций ПО, которое утверждает входные и промежуточные данные, форму выходной информации для каждой из подсистем, описывает возможное взаимодействие с другими программными комплексами

Эксплуатационные спецификации содержат сведения о быстродействии ПО, затратах памяти, требуемых технических средств, надежности и т. д.

Функциональные спецификации определяют функции, которые должны выполнять ПО, т. е. в них определяется, *что* надо делать системе, а не то, *как* это делать

Значение спецификаций:

1. Спецификация является заданием на разработку ПО и их выполнение – закон для разработчика.
2. Спецификации используются для проверки готовности ПО.
3. Спецификации являются неотъемлемой частью программной документации, облегчают сопровождение и модификацию ПО

сопровождение - это процесс исправления ошибок, координации всех элементов системы в соответствии с требованиями пользователя, внесение всех необходимых ему исправлений и изменений.

Программные документы должны содержать сведения, необходимые для разработки, изготовления, сопровождения, и эксплуатации программ. Существует следующие виды программных документов и их содержание:

- *спецификация* – состав программы и документация на неё;
- *ведомость держателей подлинников* – перечень предприятий, на которых хранятся подлинники программных документов;
- *текст программы* – запись программы с необходимыми комментариями;
- *описание программы* – сведения о логической структуре и функционировании программы;
- *программа и методика испытаний* – требования, подлежащие проверке