

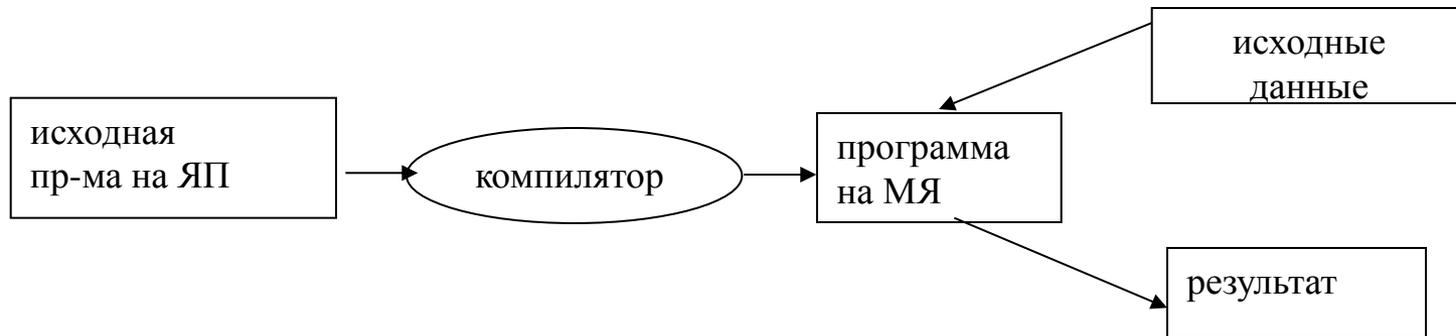
# Трансляторы

Главным компонентом систем программирования является транслятор.

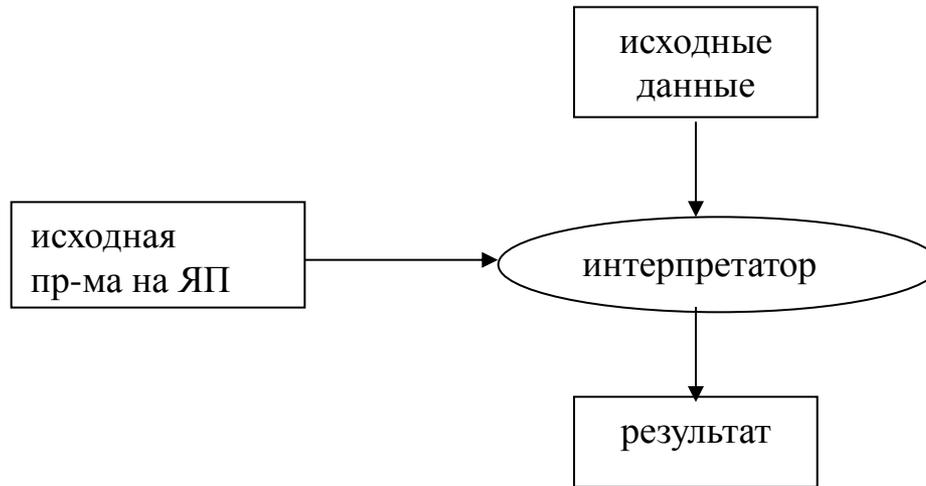
Все трансляторы подразделяются на два основных класса:

- **компиляторы,**
- **интерпретаторы.**

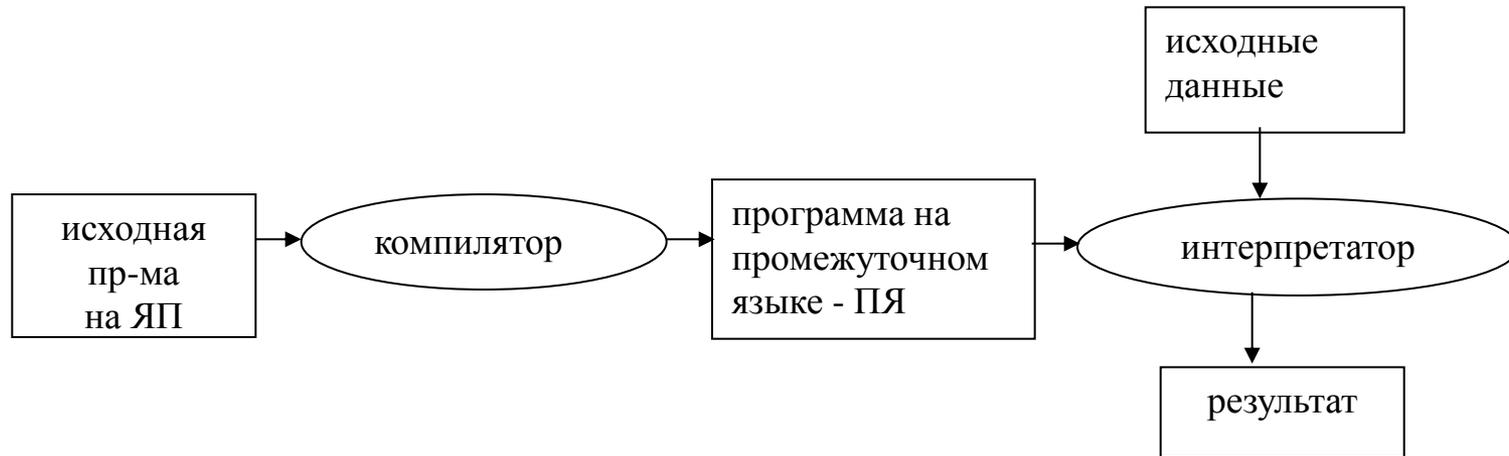
# Компилятор



# Интерпретатор



# Смешанная стратегия трансляции

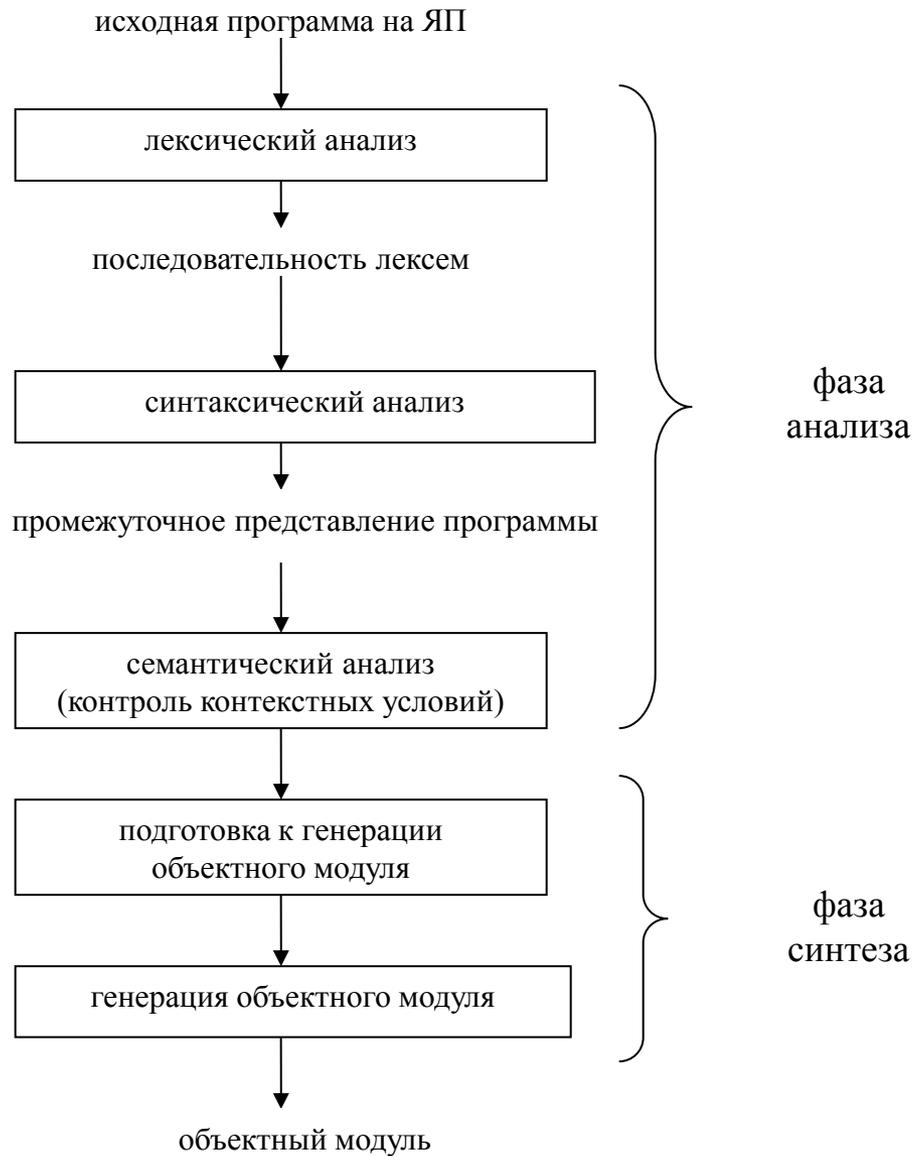


**Проход компилятора** – процесс последовательного чтения компилятором данных из внешней памяти, их обработка и помещение результата во внешнюю память, в частности, ОП.

Один проход включает в себя выполнение одного или нескольких этапов компиляции.

Результат промежуточных проходов – внутреннее представление исходной программы, результат последнего прохода – объектная программа.

# Схема функционирования компилятора



# Описание формального языка

- **Алфавит** задается перечислением конечного непустого множества символов, которые могут быть использованы для записи текстов на каком-либо языке.
- **Синтаксис** определяется набором правил, устанавливающих, какие комбинации символов алфавита являются правильными текстами на определяемом языке и позволяющих связать с каждым правильным текстом на этом языке некоторую синтаксическую структуру.
- **Семантика** определяет смысл синтаксически правильных конструкций языка, то, что означает конструкция.  
Семантика обычно описывается словами. Четкое и точное описание семантики очень важно для транслятора, т.к. его цель - получить *эквивалентную* программу на МЯ (для компилятора), либо точно выполнить указанные действия (для интерпретатора).
- **Прагматика** формального языка сводится к аргументации того, зачем та или иная конструкция вошла в состав языка.