

Языки программирования и их история

Авторы:

Илларионов Сергей Владимирович

Перлова Наталья Васильевна

Сергиев Посад

2013

Цели доклада

Многие современные люди активно пользуются компьютером, работают с различными программами, некоторые даже писали различные программы на каком-либо языке, но не знают, каким образом компьютер следует программе, какие языки программирования существуют, чем они отличаются и в чем их связь, не знают, почему на первых компьютерах работали только программисты, что такое системы программирования, что такое интерпретация и компиляция, в чем они различаются. Цель моего доклада - рассмотреть эти вопросы и навсегда устранить незнание этих вещей.

Вступление

В этом докладе я сформулирую основы знаний о языках программирования: расскажу каким образом совершенствовались языки программирования, рассмотрю язык машинных команд, ассемблеры, языки программирования высокого уровня. Представлю вам понятие уровня языков программирования, расскажу какими способами они переводятся на язык компьютера, познакомлю с системами программирования.

Основная часть

Вычислительная техника начала XX века.

Ознакомлю с вычислительной техникой начала XX века я с помощью видео. В видео шла речь о различной технике: табуляторе, счетной клавишной машинке, машине, решающей систему дифференциальных уравнений, электроинтеграторе. Взаимодействие с любой из них производилось путем физического вмешательства, т.е. управляющий руками устанавливал необходимые для машины данные.

Способность машины к выполнению определённого набора инструкций без необходимости физического вмешательства является фундаментальной особенностью компьютеров. Дальнейшее развитие эта особенность получила, когда машины приобрели способность управлять процессами путем следования заранее написанной программы. Это позволило компьютерам самостоятельно изменять порядок выполнения инструкций программы в зависимости от состояния данных.

В 1946 году в журнале "Nature" вышла статья Дж.фон Неймана, Г.Голстайна и А.Беркса "Предварительное рассмотрение логической конструкции электронного вычислительного устройства". В этой статье были изложены принципы устройства и работы ЭВМ. Главный из них - принцип хранимой в памяти программы, согласно которому данные и программа помещаются в общую память. Этот принцип позволил писать различные компьютерные программы, по которым в дальнейшем следовал компьютер. Принципиальное описание устройства и работы компьютера принято называть архитектурой ЭВМ. Идеи, изложенные в упомянутой выше статье, получили название "архитектура ЭВМ Дж.фон Неймана".

Когда появились первые ЭВМ (компьютеры), для того чтобы с ними взаимодействовать необходимо было умение писать компьютерную программу на ее

родном языке-языке машинных команд(сокращено ЯМК). В программах на ЯМК данные обозначаются их адресами в памяти машины, выполняемые операции - числовыми кодами. Программист сам должен заботиться о расположении в памяти ЭВМ команд программы и данных. Приведем пример команды на ЯМК. Она состоит из адреса операции, кода операции, адреса первого операнда, адреса второго операнда и адреса результата. Адрес операции служит для определения последовательности выполнения команд алгоритма, команды выполняются в порядке возрастания. Код операции определяет какую команду необходимо выполнить, это зависит от системы команд исполнителя (СКИ). У каждого процессора своя СКИ. Адреса операндов указывают на их расположение в оперативной памяти. Адрес результата соответственно помещается в ячейку, указанную в адресе результата. Программы на ЯМК пишутся в двоичном виде.

Со временем, в связи с трудоемкостью ЯМК, были созданы первые системы программирования, они позволяли работать на языках более высокого уровня, чем ЯМК. Первые языки более высокого уровня, чем ЯМК – автокоды (или как их позже назвали ассемблеры). Одна команда на автокоде соответствует одной команде на ЯМК. При работе с автокодом программист не распределяет память под программу, не работает с адресами ячеек, переменные и числовые константы записываются, как и в математике, коды операций буквами. Несмотря на то, что автокод -язык более высокого уровня, чем ЯМК, он все равно считается низкоуровневым языком.

Позже появились и языки, более высокоуровневые чем автокод -это языки ЯПВУ(языки программирования высокого уровня),например: Паскаль, Бейсик, Си, Фортран.

Уровень языка характеризует удаленность от языка процессора и приближенность к естественному человеческому языку, к формальному языку предметной области(чаще всего- математике). Чем выше уровень языка, тем дальше он расположен от компьютера и ближе к человеку. ЯМК считается языком самого низкого уровня. С повышением уровня языка, многие операции предусмотрены самим языком, поэтому программировать становится легче и быстрее, но теряется возможность изменять вещи, которые можно было бы изменить на более низкоуровневом языке.

Так как процессор понимает только ЯМК, любая программа в конечном итоге должна быть переведена именно на этот язык. Для этого существуют специальные программы-переводчики - трансляторы. Существуют два способа трансляции:"компиляция" и "интерпретация".

Работа компилятора

При компиляции в память компьютера загружается программа-компилятор. Она воспринимает текст программы на ЯПВУ как исходную информацию. Компилятор производит синтаксический контроль программы и при обнаружении ошибок выводит диагностические сообщения. Если ошибок нет, то результатом компиляции является программа на языке машинных команд. Затем компилятор удаляется из оперативной памяти. В памяти остается только программа на ЯМК, которая в будущем выполняется для

получения результатов.

Конечно, компиляция с автокода(ассемблера) намного проще, чем с ЯПВУ. Для этой процедуры часто применяют специальный термин-ассемблирование. А под словом "ассемблер" понимается не только язык программирования, но и транслятор с него.

Работа интерпретатора

Интерпретатор в течение всего времени работы программы находится во внутренней памяти. В ОЗУ помещается программа на ЯПВУ. Интерпретатор читает ее первый оператор, переводит его в машинные команды и тут же организует выполнение этих команд. Затем переходит к переводу и выполнению следующего оператора и так до конца программы. При этом результаты предыдущих переводов в памяти не сохраняются. Перед трансляцией каждого оператора происходит его синтаксический анализ.

Системы программирования

Система программирования - комплекс инструментальных программных средств, предназначенный для работы с программами на одном из языков программирования. Первые СП включали в себя только транслятор. Затем к нему стали добавляться различные сервисные средства, например, текстовые редакторы, средства организации взаимодействия с пользователем и т.д. Язык программирования, с которым работает СП называется ее входным языком. По своему внешнему виду она во многом похожа на текстовый редактор.

Заключение

Когда появилась возможность записывать программу, чтобы компьютер в дальнейшем выполнял ее, начали развиваться языки программирования. Языки программирования делятся на две категории: языки программирования низкого и высокого уровня. Уровень языка характеризует его приближенность к человеческому языку. Низкоуровневые языки в свою очередь делятся на язык машинных команд и ассемблеры(автокоды).Но, так как процессор воспринимает только язык машинных команд, любая программа, написанная на каком-либо другом языке, в конечном итоге переводится на ЯМК. Для этого существуют специальные программы-переводчики - трансляторы. Трансляторы делятся на два вида: интерпретаторы и компиляторы. Интерпретаторы переводят программу по одному оператору и сразу выполняют его. Компиляторы сначала переводят всю программу, а пользователь за тем сам выбирает когда ей воспользоваться, также он может сохранить перевод и пользоваться им в дальнейшем. Для удобства программистов были созданы системы программирования, сначала они включали в себя только транслятор, но позже включили различные средства, например текстовые редакторы. Системы программирования сильно облегчают труд программиста, предоставляют отличную визуализацию программы.

Список источников информации

И.Семакин, Л.Залогова, С.Русаков, Л.Шестакова. О языках программирования и трансляторах. -Информатика 9,2005. - 341 с.

Свободная энциклопедия - «Википедия».

Изображения - «Google Картинки».

Представленный документальный видеосюжет является частью фильма об открытии комплекса зданий МГУ на Ленинских горах (ныне Воробьёвых) и фильма об академике Сергее Львовиче Соболеве (1908—1989) — в то время заведующего кафедрой вычислительной математики.

Оглавление

Цели доклада	2
Вступление	2
Основная	2
Работа компилятора	3
Работа интерпретатора	4
Системы программирования	4
Заключение	4
Список источников информации	5