

СОЗДАНИЕ МНОГОУРОВНЕВОГО КОМПЬЮТЕРА И ПОНИМАНИЕ ЯЗЫКОВ, УРОВНЕЙ И ВИРТУАЛЬНЫХ МАШИН

Тожибоев Иброхимжон Тоджалиевич

Ферганский филиал ТУИТ имени Мухаммада ал-Хоразмий

Заместитель директора по учебной работе

Мамирходжаев Мухаммадамин Мавлонжонович

Ганиев Санжарбек Фазлиддинович

Умаралиев Джамшидбек Тохтасинович

Студенты Ферганского филиала ТУИТ имени Мухаммада ал-Хоразмий

Аннотация: статья дает представление о порядке и типах машин высокого уровня, о важной взаимосвязи между языком и виртуальной машиной, а также о многих сходствах между переводом и интерпретацией.

Ключевые слова: виртуальные машины, машины уровней, машинный язык.

CREATING A MULTI-LEVEL COMPUTER AND UNDERSTANDING LANGUAGES, LEVELS AND VIRTUAL MACHINES

Tojiboev Ibrokhimjon Tojalievich

Mamirkhodzhaev Mukhammadamin Mavlonzhonovich

Ganiev Sanjarbek Fazliddinovich

Umaraliev Jamshidbek Tokhtasinovich

Abstract: The article provides insight into the order and types of high-level machines, the important relationship between language and a virtual machine, and the many similarities between translation and interpretation.

Key words: virtual machines, level machines, machine language.

Цифровой компьютер - это машина, которая может решать проблемы, выполняя данные ему команды. Последовательность команд, описывающих решение конкретной проблемы, называется программой. Схема каждого

компьютера распознает и может выполнять ограниченный набор простых команд. Перед выполнением все программы обычно необходимо преобразовать в простую несложную последовательность команд, например:

- ✓ Сложите два числа
- ✓ Убедитесь, что число не равно нулю;
- ✓ Скопируйте блок данных из одной части памяти компьютера в другую.

Этот базовый набор простых команд - это язык, на котором люди могут общаться с компьютером. Такой язык называется машинным. При создании нового компьютера вам необходимо решить, какие команды вводить на машинном языке этого компьютера.

Выше указанную проблему можно решить двумя способами. Оба метода предполагают разработку нового программного обеспечения. Этот новый набор команд создает язык, который мы называем T1. Встроенные машинные команды также создают язык, который мы называем T0. Компьютер может запускать только программы, написанные на языке программирования T0. Два способа решения проблемы различаются тем, как компьютер выполняет программы, написанные на языке T1, однако компьютер имеет только машинный язык T0.

Первый способ выполнить программу, написанную в T1, - это заменить каждую команду соответствующим набором команд в T0.

В этом случае компьютер выполняет новую программу, записанную в T0, вместо старой программы, записанной в T1. Эта технология называется трансляцией.

Второй метод - создать программу, которая принимает программы, написанные в T1, в качестве входных данных в T0. В этом случае каждая команда на языке T1 обрабатывается по очереди, после чего немедленно выполняется соответствующий набор команд на языке T0. Эта технология не требует создания новой программы в программе T0. Он называется

интерпретатором, а программа, которая его выполняет, называется интерпретатором.

Между трансляцией и устным переводом много общего. В обоих случаях компьютер выполняет набор команд T0, эквивалентных командам T1. Единственное отличие состоит в том, что во время трансляции программа T1 преобразуется в программу T0, программа T1 удаляется, а новая программа T0 загружается в память компьютера и затем выполняется. Во время работы сгенерированная программа, написанная на языке T0, управляет работой компьютера. При интерпретации каждая команда в T1 перезаписывается на T0 и выполняется немедленно. В этом случае программа вещания не создается.

Однако гораздо проще представить себе существование гипотетического компьютера или виртуальной машины с машинным языком T1, чем думать о переводе и интерпретации. Мы называем такую виртуальную машину M1, а виртуальную машину, работающую на языке T0, - M0. Если бы такую машину M1 можно было построить без больших затрат, не было бы необходимости в машине, которая выполняет программы в T0 и T0. То есть программы могут быть написаны на T1, и компьютер может их немедленно выполнить. Возможно, создать виртуальную машину невозможно (потому что это слишком дорого или сложно построить), но люди могут писать программы, ориентированные на нее.

Если языки T0 и T1 не сильно отличаются друг от друга, рекомендуется перевод и устный перевод. Во многих случаях это означает, что T1 лучше, чем T2, но все же далек от идеала. Возможно, настоящая цель создания этого языка T1 - избавить программиста от бремени написания программ на языке, который понимает компьютер, но не адаптирован для людей. Однако ситуация не так безнадежна.

Очевидное решение проблемы - создать другой набор команд, более ориентированный на человека и менее ориентированный на компьютер, чем T1. Этот третий набор команд создает язык, который мы называем T2, и

соответствующую виртуальную машину M2. Программист может писать программы на T2, как если бы есть виртуальная машина для работы с машинным языком T2. Такие программы могут быть переведены на T1 или выполнены интерпретатором, написанным на T1.

Изобретение разных языков может продолжаться до тех пор, пока мы не придем к языку, который больше подходит для каждого из нас, чем предыдущий. Каждый из этих языков использует в качестве основы своего предшественника, поэтому мы называем компьютер можно рассматривать как серию пораженных поверхностей на рисунке 1.1. Язык в нижней части иерархической структуры самый простой, а в верхней - самый сложный.

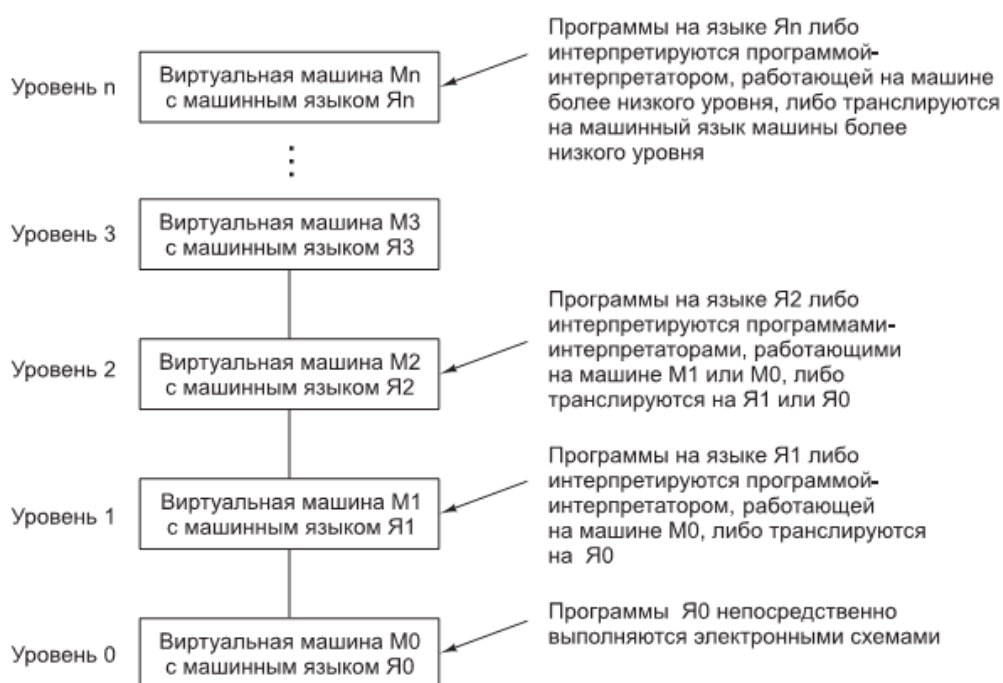


Рисунок 1.1. Многоступенчатая машина

Между языком и виртуальной машиной существует важная взаимосвязь. Каждая машина поддерживает определенный машинный язык, который содержит все команды, которые машина может выполнять. Фактически, машина определяет язык. Точно так же язык определяет машину, то есть машину, которая выполняет все программы, написанные на этом языке.

Компьютер уровня n можно представить как n различных виртуальных машин, каждая со своим собственным машинным языком. Мы используем термины «уровень» и «виртуальная машина» как синонимы. Только программы, написанные на T_0 , могут выполняться без компьютерного перевода или интерпретации. Программы, написанные на T_1, T_2, \dots, T_n , должны проходить через интерпретатор более низкого уровня или переводиться на язык более низкого уровня.

Человеку, который пишет программы для виртуальной машины n -го уровня, не нужно знать об интерпретаторах и интерпретаторах более низкого уровня.

Большинство программистов, использующих машины n -го уровня, интересуются только машинным языком самого высокого уровня. Однако тем, кто хочет понять, как на самом деле работает компьютер, следует изучить все основы. Разработчики новых компьютеров или новых уровней (т.е. новых виртуальных машин) должны быть знакомы со всеми уровнями.

РЕКОМЕНДАЦИИ

1. Р.Р. Бокиев, А.О. Матчанов, Информатика, «Учитель», Ташкент, 2010, 374 с.
2. Сатторов А. Информатика и информационные технологии, Учебник, Ташкент, 2013, 320 стр. 159
3. С.С. Гуломов и другие. Информационные системы и технологии. Учебник. Ташкент, Шарк, 2000.