

Российский университет дружбы народов
Факультет физико-математических и естественных наук
Кафедра прикладной информатики и теории вероятности

Презентация на тему:
Ассемблер NASM. Основные команды, типы данных, структура программы

Выполнил: студент Матюшкин Д.В. (НПИбд-02-21)
Преподаватель: Демидова Анастасия Вячеславовна



Терминология

Язык ассемблера (англ. *assembly language*) — машинно-ориентированный язык программирования низкого уровня. Представляет собой систему обозначений, используемую для представления в удобно читаемой форме программ, записанных в машинном коде.

Перевод программы на языке ассемблера в исполнимый машинный код (вычисление выражений, раскрытие макрокоманд, замена мнемоник собственно машинными кодами и символьных адресов на абсолютные или относительные адреса) производится **ассемблером** — программой-транслятором, которая и дала языку ассемблера его название.

NASM (Netwide Assembler) — свободный ассемблер для архитектуры Intel x86. Используется для написания 16-, 32- и 64-разрядных программ.

Синтаксис языка ассемблер

В NASM используется Intel-синтаксис записи инструкций. Предложение языка ассемблера NASM (строка программы) может состоять из следующих элементов:

[метка:] команда/директива [операнд {, операнд}] [; комментарий]

Операнды отделяются между собой запятой. Перед строкой и после инструкции можно использовать любое количество пробельных символов. Комментарий начинается с точки с запятой, а концом комментария считается конец строки. В качестве инструкции может использоваться команда или псевдокоманда (директива компилятора). Если строка очень длинная, то её можно перенести на следующую, используя обратный слеш \, подобно тому, как это делается в языке Си.

- Команда - указывает транслятору, какое действие должен выполнить микропроцессор.
- Операнд – объект, над которым выполняется машинная команда.
- Метка - допустимыми символами в метках являются буквы, цифры, знаки `_`, `$`, `#`, `@`, `~`, `.` и `?`.
- Комментарии в ассемблере должны начинаться с точки с запятой (`;`). Они могут содержать любой печатный символ, включая пробел. Комментарий может находиться на отдельной строке.

Примеры строк кода:

Count	db	1	;Имя, директива, один операнд
mov	eax,0		;Команда, два операнда
cbw			; Команда

Основные команды:

Типичными командами языка ассемблера являются (большинство примеров даны для Intel-синтаксиса архитектуры x86):

- Команды пересылки данных (mov и др.)
- Арифметические команды (add, sub, imul и др.)
- Логические и побитовые операции (or, and, xor, shr и др.)
- Команды управления ходом выполнения программы (jmp, loop, ret и др.)
- Команды вызова прерываний (иногда относят к командам управления): int
- Команды ввода-вывода в порты (in, out)

Создание и обработка программы

Структура программы NASM



Программы на ассемблере могут быть разделены на три секции:

- **секция data:**
section.data
- **секция bss:**
section.bss
- **секция text:**
section.text
global _start
_start:

Заключение

На языке ассемблера пишут:

- Программы, требующие максимальной скорости выполнения
- Программы, взаимодействующие с внешними устройствами
- Программы, использующие полностью возможности процессора
- Программы, полностью использующие возможности операционной системы

Достоинства языка ассемблера:

- Максимальная оптимизация программ;
- Максимальная адаптация под соответствующий процессор.

Недостатки языка ассемблера:

- Трудоемкость написания программы больше, чем языке высокого уровня;
- Непереносимость на другие платформы, кроме совместимых.

Список источников:

1. https://ru.m.wikipedia.org/wiki/Язык_ассемблера
2. <https://ru.wikipedia.org/w/index.php?title=NASM&stable=1>
3. https://esystem.rudn.ru/pluginfile.php/1185223/mod_resource/content/0/Лабораторная%20работа%202.pdf
4. <https://prog-cpp.ru/asm/>
5. <https://ravesli.com/assembler-bazovyj-sintaksis/>

Спасибо за внимание!

