PA1 实验报告

必答题

理解基础设施

1. 那么这个学期下来, 你将会在调试上花费多少时间?

$$500 * 90\% * 30 * 20 = 270000s$$

2. 那么这个学期下来, 简易调试器可以帮助你节省多少调试的时间?

$$270000/2 = 135000s$$

查阅i386手册

1. EFLAGS寄存器中的CF位:

   Page 33 of 421（2.3.4 Flags Register）

   > "Refer to Appendix C for definition of each status flag."

   Page 419 of 421

   > Status Flags' Functions
   >
   > Bit Name Function
   >
   > 0 CF Carry Flag —— Set on high-order bit carry or borrow; cleared otherwise.

2. ModR/M字节:

   Page 241 of 421

   > The ModR/M and SIB bytes follow the opcode byte(s) in many of the 80386 instructions. They contain the following information:
   >
   > - The indexing type or register number to be used in the instruction
   > - The register to be used, or more information to select the instruction
   > - The base, index, and scale information
   >
   > The ModR/M byte contains three fields of information:
   >
   > - The mod field, which occupies the two most significant bits of the byte, combines with the r/m field to form 32 possible values: eight registers and 24 indexing modes
   > - The reg field, which occupies the next three bits following the mod field, specifies

> either a register number or three more bits of opcode information. The meaning of the reg field is determined by the first (opcode) byte of the instruction.
> - The r/m field, which occupies the three least significant bits of the byte, can specify a register as the location of an operand, or can form part of the addressing-mode encoding in combination with the field as described above

3. mov指令的具体格式：

Page 345 of 421 ~ Page 351 of 421

shell命令

1. nemu/ 目录下的所有.c和.h和文件总共有多少行代码?

   答：3223

   命令：`$ find . -name "*.[hc]"|xargs cat|grep -v ^$|wc -l`

2. 和框架代码相比, 你在PA1中编写了多少行代码?

   $pa1 - pa0 = 3223 - 2813 = 410$行

3. 使用man 打开工程目录下的 Makefile 文件, 你会在 CFLAGS 变量中看到gcc的一些编译 选项. 请解释gcc中的 -Wall 和 -Werror 有什么作用? 为什么要使用 -Wall 和 -Werror?

   1. -Wall的作用

      > -Wall This enables all the warnings about constructions that some users consider questionable, and that are easy to avoid (or modify to prevent the warning), even in conjunction with macros. This also enables some language-specific warnings described in C++ Dialect Options and Objective-C and Objective-C++ Dialect Options.

      使用原因：避免一些questionable的naive的语法问题。

   2. -Werror的作用

      > -Werror Make all warnings into errors.

      使用原因：防止在有warning的情况下仍然运行程序。