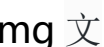



PA_2-1 实验报告

计算机科学与技术系 张桓 191220156

1. 使用 `hexdump` 命令查看测试用例的文件，所显示的文件的内容对应模拟内存的哪一个部分？指令在机器中表示的形式是什么？

答：①测试用例的镜像文件内容对应于模拟内存中 0x30000 起开始的连续区域部分。

②指令在机器中的表示形式是 01 串。

2. 如果去掉 `instr_execute_2op()`函数前面的 `static` 关键字会发生什么情况？为什么？

答：`static` 关键字的作用是限制函数只能在当前声明它的文件中使用，同时不同的文件可以使用相同名字的静态函数。去掉 `static` 关键字会导致各个用 `make_instr_impl_2op` 宏实现的指令文件产生同名函数冲突。因为用 `make_instr_impl_2op` 实现的指令都会调用 `instr_execute_2op()`函数，为了实现对应指令的功能每个指令文件的 `instr_execute_2op()`都是不同的，为了只在当前指令文件中调用必须添加 `static` 关键字限制范围，否则会与其他也用到 `make_instr_impl_2op` 实现的指令造成冲突，这样做实现了不同指令都可以用同一套宏来完成。

3. 为什么 `test-float` 会 `fail`? 以后在写和浮点数相关的程序的时候要注意什么?

答: 最后的浮点数运算, 用 `float` 形式表示

`1.2=00111111100110011001100110011010`

`1.0=00111111100000000000000000000000`

`2.2=01000000000011001100110011001101`

`0.2=00111110010011001100110011001101`

`1.2+1.0` 右归 1 次后的结果和 `2.2` 比较中间结果二者是相等的, 所以没有触发第一个 `nume_trap`。

而 `1.2-1.0` 左归 3 次的结果 `0.2` 无法用二进制精确表示中间结果:

`00111111100110011001100110011010` 无法相等。

我们以后写浮点数相关程序时一定要注意精度问题, 因为比较是直接比较机器数是否相等, 而由于浮点数运算的左归右归的问题很容易造成精度损失导致误判。