CS 241 – Week 5 Tutorial

MERL

Spring 2015

Summary

- MERL to ASM translation
- Unlinking MERL files

Problems

1. A file called main.asm has been assembled with `java cs241.linkasm` to produce main.merl. Using `xxd` we can see the contents of main.merl.

```
0000000: 1000 0002  ....
0000004: 0000 0060  ...‘
0000008: 0000 0020  ...
000000c: 0000 001c  ....
0000010: 0000 0014  ....
0000014: 0000 0014  ....
0000018: 0000 0020  ...
000001c: 0000 0000  ....
0000020: 0000 0001  ....
0000024: 0000 0014  ....
0000028: 0000 0001  ....
000002c: 0000 000c  ....
0000030: 0000 0011  ....
0000034: 0000 001c  ....
0000038: 0000 0003  ....
000003c: 0000 0064  ...d
0000040: 0000 0065  ...e
0000044: 0000 0066  ...f
0000048: 0000 0005  ....
000004c: 0000 0014  ....
0000050: 0000 0003  ....
0000054: 0000 0061  ...a
0000058: 0000 0062  ...b
000005c: 0000 0063  ...c
```

   Due to poor file management we have lost main.asm and want to recreate using using main.merl. `printmerl` is a useful tool for printing out the contents of MERL files in a human readable form. However, for this exercise, assume we don't have access to `printmerl`.

First translate main.merl into a format similar to what `printmerl` would produce and then, using this translation, produce an equivalent asm file to the original main.asm. (Why can't we reproduce the exact original main.asm?)

2. Now we have linked another merl file, lib.merl, with main.merl (in that order) to produce combined.merl. Still having not learned our lesson about file management we have again lost lib.merl (along with the asm source for it, lib.asm). We want to recreate both lib.merl and lib.asm. Luckly we now have access to `printmerl` and using it can see the contents of combined.merl:

```
cookie   10000002
length         90
clen           30
0000000c       24
00000010       18
00000014       14
00000018       24
0000001c       2c
00000020       14
00000024       24
00000028       20
0000002c       10
REL            10
REL            1c
REL            24
REL            2c
REL             c
REL            18
ESD            24 abc
ESD            10 def
```

First produce what `printmerl` would produce if given lib.merl, then take that output and produce an equivalent to the original lib.asm.