Problem Set 0 -- Comp Arch / C Language Review

1) Describe at least three operand addressing modes.

2) What is a CPU "register?" What circuit technology is used to implement it and how does that differ from DRAM?

3) What is a cache (specifically, a data cache)? Why is it used?

4) What is an interrupt handler aka interrupt service routine? What causes it to be invoked?

5) All computer architectures have a register which is generically called the "program counter." How do we interpret the value in that register?

6) Likewise (almost) all architectures have a "stack pointer" register. What does its value represent? When does that value change?

7) "ROM" stands for read-only memory. Why would we want a memory that we can’t write to, only read?

8) Some hexadecimal math, you should be able to do this in your head!

a) -9 as a signed, 32 bit integer in HEX

b) (signed int) -5 >> 1 (express in HEX and decimal)

c) (unsigned int) 1024 | 0x0A (express in HEX and decimal)

d) (unsigned int) 0x678 & ~255 (express in HEX and decimal)

e) 'e' ^ 32 (express in HEX, decimal, and as a character)

9) The following fragment of a C program attempts to read a line of input and echo it back, but is incorrect. Explain why, and offer a suggestion to fix it.

```
void somefunction(void)
{
 char *s;
          printf("Type something\n");
          scanf("%s",s);
          printf("You typed %s\n",s);
}
```

10) What does this program print out and why?

```
int main(int argc, char **argv)
{
 int i,r,fn();
          for(i=0;i<10;i++)
                    r=fn();
          printf("%d\n",r);
}

int fn(void)
{
 static int s=5;
          return s++;
}
```