Homework Assignment 2

(Due 2:10pm, Oct. 5, email to daehyun.kim@wsu.edu or submit a hardcopy)

You can use the following instructions only for this homework.

- Instructions
  - ADD R$, R%, R&
  - ADD R$, R%, #imm
  - SUB R$, R%, R&
  - SUB R$, R%, #imm
  - AND R$, R%, R& // logical AND
  - AND R$, R%, #imm
  - ORR R$, R%, R& // logical OR
  - ORR R$, R%, #imm
  - EOR R$, R%, R& // logical XOR
  - EOR R$, R%, #imm
  - CMP R$, R%
  - CMP R$, #imm
  - BGE, BLT, BGT, BLE, BEQ, BNE, B
  - MOV R$, R%// R$ = R%
  - MOV R$, #imm
  - MOR R$, R%, LSL #imm (or LSR #imm)

1. (30 points) Write an assembly code for the following C code.

```
int a, b, c;

switch ( a ) {
  case 0: b++; break;
  case 1: b--; break;
  case 2: c++; break;
  default: b = 0; c = 1; break;
}
```

- Assume that a is in R0, b is in R1, and c is in R2.
- The exit point (the end of the code) could be just an address label.

2. (40 points) Write an assembly code for the following C code.

```
int a, b, c;

for ( a = 0 ; (a / 4) < 10 ; a = a + 2 ) {
  b++;

  if ( b < c )
    b++;
  else
    c++;

  if ( (c % 4) == 1 )
    break;
}
```

- Assume that a is in R0, b is in R1, and c is in R2.
- The exit point (the end of the code) could be just an address label.

3. (50 points) Write an assembly code for the following C code.

```
int a, b, c;

a = 1;
b = 2;
c = 3;

while ( a < 10 ) {
  while ( b < 20 ) {
    if ( c < 30 )
      c++;
    else
      c += 2;

    b++;
  }
  a++;
}
```

- Assume that a is in R0, b is in R1, c is in R2, and n is in R3.
- The exit point (the end of the code) could be just an address label.