# Homework 1 Solution

## Problem 1

```
int Sec_big= [
    (A<=B && C<=A && D<=A)||(A<=C && B<=A && D<=A)||(A<=D && B<=A && C<=A)
                                                                    : A;
    (B<=A && C<=B && D<=B)||(B<=C && A<=B && D<=B)||(B<=D && A<=B && C<=B)
                                                                    : B;
    (C<=A && B<=C && D<=C)||(C<=B && A<=C && D<=C)||(C<=D && A<=C && B<=C)
                                                                    : C;
   1                                                                : D;
];
```

## Problem 2

| **Stage** | Generic | Specific |
|---|---|---|
| | `rrmovl rA, rB` | `rrmovl %esp,%ebp` |
| **Fetch** | icode : ifun ← M1[PC]<br>rA:rB←M1[PC+1]<br><br>valP←PC+2 | icode : ifun ← M1[0x044] = 2 : 0<br>rA:rB←M1[0x045] = 4 : 5<br><br>valP←0x044+2 = 0x046 |
| **Decode** | valA ← R[rA] | valA ← R[%esp] = 128 |
| **Execute** | valE ← 0 + valA | valE ← 128 |
| **Memory** | | |
| **Write Back** | R[rB] ← valE | R[%ebp] ← 128 |
| **Update PC** | PC ← valP | PC ← 0x046 |

## Problem 3

| **Stage** | Generic | Specific |
|---|---|---|
| | `OPl rA, rB` | `xorl %eax,%eax` |
| **Fetch** | icode : ifun ← M1[PC]<br>rA:rB ← M1[PC+1]<br><br>valP ← PC+2 | icode : ifun ← M1[0x052] = 6 : 3<br>rA:rB ← M1[0x053] = 0 : 0<br><br>valP ← PC+2 = 0x054 |
| **Decode** | valA ← R[rA]<br>valB ← R[rB] | valA ← R[%eax] = 1<br>valB ← R[%eax] = 1 |
| **Execute** | valE ← valB OP valA<br>Set CC | valE ← 1 ^ 1 = 0<br>ZF ← 1, SF ← 0, OF ← 0 |
| **Memory** | | |
| **Write Back** | R[rB] ← valE | R[%eax] ← 0 |
| **Update PC** | PC ← valP | PC ← 0x054 |

## Problem 4

| **Stage** | Generic | Specific |
|---|---|---|
| | `irmovl V, rB` | `irmovl $-1,%ebx` |
| **Fetch** | icode : ifun ← M1[PC]<br>rA:rB ← M1[PC+1]<br>valC ← M4[PC+2]<br>valP ← PC+6 | icode : ifun ← M1[0x06b] = 3 : 0<br>rA:rB ← M1[0x06c] = f : 3<br>valC ← M4[0x06d] = -1<br>valP ← 0x06b+6 = 0x071 |
| **Decode** | | |
| **Execute** | valE←0+valC | valE←0+(-1) = -1 |
| **Memory** | | |
| **Write Back** | R[rB] ← valE | R[%ebx] ← -1 |
| **Update PC** | PC ← valP | PC ← 0x071 |

## Problem 5

| **Stage** | Generic | Specific |
|---|---|---|
| | `OPl rA, rB` | `addl %ebx,%edx` |
| **Fetch** | icode : ifun ← M1[PC]<br>rA:rB ← M1[PC+1]<br><br>valP ← PC+2 | icode : ifun ← M1[0x071] = 6 : 0<br>rA:rB ← M1[0x072] = 3 : 2<br><br>valP ← 0x071+2 = 0x073 |
| **Decode** | valA ← R[rA]<br>valB ← R[rB] | valA ← R[%ebx] = -1<br>valB ← R[%edx] = 2 |
| **Execute** | valE ← valB OP valA<br>Set CC | valE ← 2 + (-1) = 1<br>ZF ← 0, SF ← 0, OF ← 0 |
| **Memory** | | |
| **Write Back** | R[rB] ← valE | R[%edx] ← 1 |
| **Update PC** | PC ← valP | PC ← 0x073 |