Michael Masenheimer CSC 252 HW 2

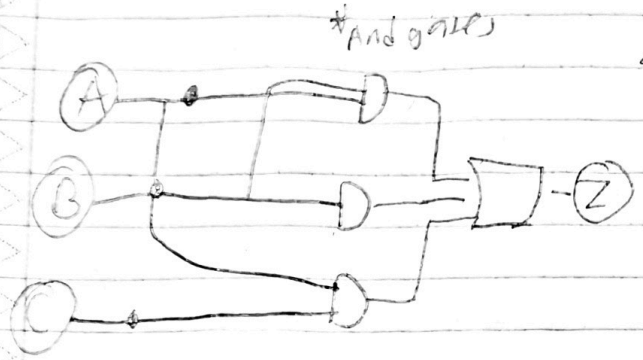
2.1 (i)
```
la $s4, bnsf    # load bnsf
lw $s4, 0($s4)

sub $t0, $s4, $s2    # bnsf-atsf
sub $t1, $s1, $s0    # csx-kcs
slt $t2, $t1, $t0    # prev > prev

bne $t2, $zero, EXIT
la $t3, $t0
lw $t3, $t4          # Store bnsf-atsf

EXIT:
  # rest of code
```

2.1 (ii)
```
la $s4, erie
lw $s4, 0($s4)
slt $t0, $s0, $s4    # erie > kcs

slt $t1, $s4, $s5    # erie < epsw

bne $t1, $zero, EXIT

bne $t0, $zero, EXIT

la $t5, $t6          # kcs into epsw
lw $t5, 0($t6)

EXIT:

# rest of code
```

2.2(c)

```
.data
mask .word 0x55555555

.text
    la $t0, mask        # load address of mask
    lw $t0, 0($t0)      # load mask into $t0
    and $t1, $s0, $t0   # apply the mask
```

2.3(d)



$Z = (A \wedge \neg C) \vee (\neg B \wedge \neg C) \vee (\neg A \wedge B \wedge C)$

$X = A'B'C' + A'BC + AB'C' + ABC' + ABC$

$Y = A'B'C + A'B'C' + ABC$

2.4(d)

```
    li $t1, 1
    li $t0, 0

LOOP:
    bge $t0, $s0, DONE   # i < row
    sll $s1, $s1, 1      # prod *= 2
    addi $t0, $t0, 1     # $t0++

DONE:
    # rest of code
```