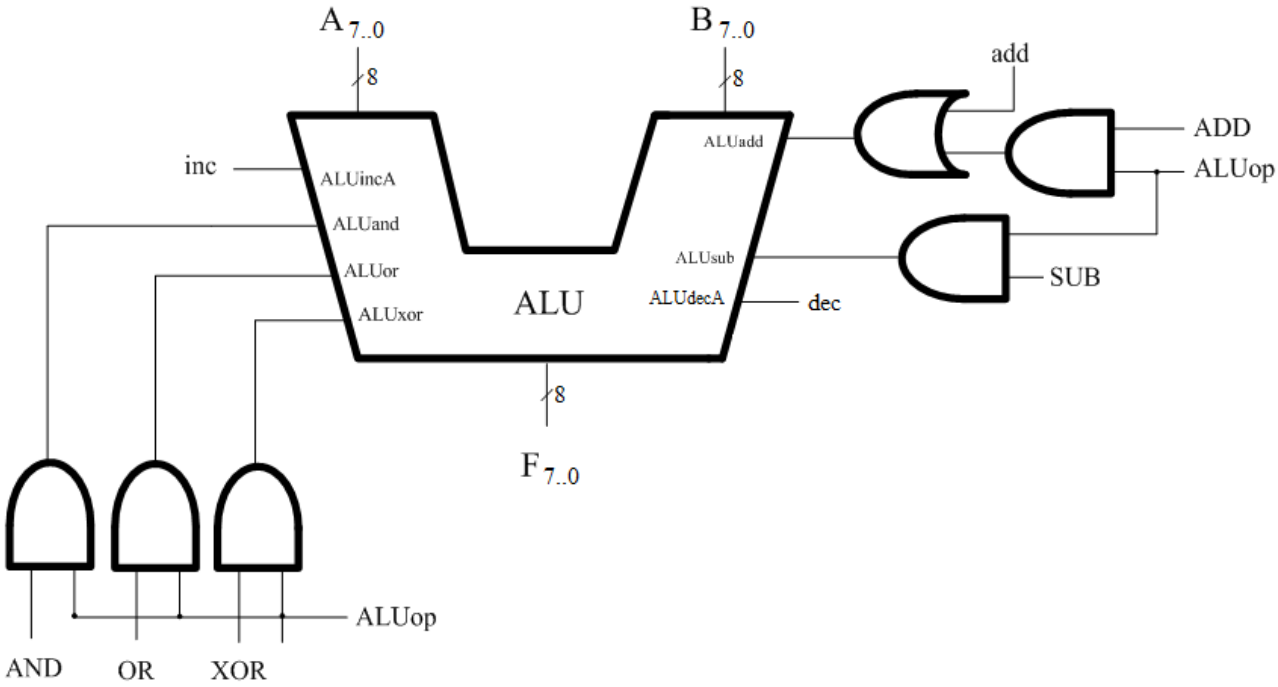
# Ispit iz Arhitekture i organizacije računara 2

a) (5p)




b) (20p)

```
; Dohvatanje instrukcije
BEGIN:      PCout,MARin,Xin
            read,incA,ALUout,PCin
            wmfc
            MDRout,IRin
; Dekodovanje instrukcije
            opcase
; binarne instrukcije (ADD, SUB, AND, OR, XOR)
BIN:        admodbin          ; način adresiranja

; PC-relativno adresiranje
BINPCR:     PCout,MARin,Xin
            read,incA,ALUout,PCin,Xin
            wmfc
            MDRout,Yin
            add,ALUout,MARin
            read
            wmfc
            MDRout,Yin
            branch(,BINOP)

; Registarsko direktno adresiranje
BINRD:      REGout,Yin
            branch(,BINOP)

; Memorijsko direktno adresiranje
BINMD:      PCout,MARin,Xin
            read,incA,ALUout,PCin
            wmfc
            MDRout,MARin
```

```
                read
                wmfc
                MDRout,Yin
                branch(,BINOP)

; Memorijsko indirektno adresiranje
BINMI:          PCout,MARin,Xin
                read,incA,ALUout,PCin
                wmfc
                MDRout,MARin
                read
                wmfc
                MDRout,MARin
                read
                wmfc
                MDRout,Yin

; Izvrsavanje
BINOP:          Aout,Xin
                ALUop,ALUout,Ain,ldPSW
                branch(IRR,INTH)
                branch(,BEGIN)
```

c) (5p)

```
START:      LOAD        R1          ; A:=R1 pocetna adresa niza
            STORE       FFh         ; mem[FFh]:=A pocetna adresa niza
            LOAD        R0          ; A:=R0
            AND         R0          ; A:=A AND R0 provera da li je 0
            JZ          END         ; if(R0=0)goto end
LOOP:       LOAD        (FFh)       ; A:=a[i]
            SUB         50h         ; A:=A + mem[50h]
            STORE       (FFh)       ; a[i]:=A
            LOAD        FFh         ; A:=mem[FFh]
            ADD         R2          ; A:=A + R2:=A + 1
            STORE       FFh         ; mem[FFh]:=A
            LOAD        R0          ; A:=R0
            SUB         R2          ; A:=A - R2:=A - 1
            STORE       R0          ; R0:=A
            JNZ         LOOP        ; NEXT
END:
```