

Objasni Funkcionalnost Instrukcija LDA, STA i TAX: Što ove instrukcije rade?

- LDA Instrukcija učitava zadanu vrijednost u akumulator.
- STA Instrukcija sprema vrijednost iz akumulatora u zadanu memoriju.
- TAX Instrukcija kopira vrijednost iz akumulatora u X registar.

Napiši instrukciju koja postavlja registar A na heksadekadsku vrijednost \$FF.

```
LDA #$FF
```

Napiši kratak program za učitavanje vrijednosti \$07 u registar X, a zatim pohrani ovu vrijednost na lokaciju nulte stranice na \$20.

```
LDX #$07  
STX $0020
```

(Nulta stranica počinje sa 00, prva stranica sa 01 itd.. Zato se piše prve dve nule u ovom zadatku)

Napiši dio skupa instrukcija koje se granaju na oznaku SKIP ako je vrijednost u registru A nula.

```
CMP #$00  
BEQ SKIP  
SKIP:
```

(ChatGPT)

Prikazi kako gurnuti registar A na stog, a zatim ga povući natrag sa stoga.

```
Guranje vrijednosti iz registra A na stog  
PHA  
Povlačenje vrijednosti s vrha stoga u registar A  
PLA
```

Napiši jednostavnu podrutinu koja učitava akumulator s vrijednošću \$10 i zatim se vraća iz podrutine.

```
ImePodrutine:  
LDA #$10  
RTS
```

Izvedi bitovni AND između akumulatora koji sadrži \$FF i neposredne vrijednosti \$0F. Koji je rezultat u akumulatoru?

```
LDA #$FF  
AND #$0F
```

Rezultat u akumulatoru je \$0F

Napiši instrukciju za brisanje zastavice Carry

```
CLC
```

Napiši petlju koja smanjuje registar Y s \$05 na \$00, smanjujući za 1 u svakom koraku.

```
LDY #$05  
LOOP:  
DEY
```

CPY # $\$00$
BNE LOOP

(DEY – Dekremenitra (smanjuje) broj)

(BNE – Branch if not Equal završava petlju kada CPY# $\$00$ NE dobije rezultat 00 na Y)

Napiši dio programa koji skoči na adresu $\$C000$ ako zastavica nula nije postavljena.

```
BNE SkipToC000
SkipToC000:
    JMP  $\$C000$ 
```

(ChatGPT)

Usporedi registar X s vrijednošću $\$10$ i granaj na oznaku EQUAL ako su jednaki.

```
CPX # $\$10$ 
BEQ EQUAL
EQUAL:
```

(ChatGPT)

Dodaj vrijednost $\$01$ akumulatoru kada sadrži $\$FF$. Što se događa sa zastavicom C?

$\$00$, carry flag se postavlja na 1

Napiši instrukciju za učitavanje akumulatora s vrijednošću pohranjenom na memorijskoj adresi

$\$0300$ I iskoristi offset (pomak) u registru X. Pomak neka bude 5

Prikazi kako koristiti neizravni način adresiranja za učitavanje vrijednosti u registar A.

```
LDA # $\$00$ 
STA  $\$10$ 
LDA # $\$02$ 
STA  $\$11$ 
LDY # $\$03$ 
LDA ( $\$10$ ), Y
```

Objasni prednost korištenja adresiranja nulte stranice nad apsolutnim adresiranjem.

Zero page adresiranje koristi samo $\$00$ do $\$FF$, brže je i ne koristi velike byte.

Pretvori binarnu vrijednost $\%10101010$ u heksadecimalnu i napiši skup instrukcija za učitavanje ove vrijednosti u akumulator

```
 $\$AA$ , LDA # $\$AA$ 
```