

Hardverközei Programozás 1

MINTA zárthelyi dolgozat (2015)

MINTA

Név: _____

Neptun: _____

- 1.) Végezd el a következő számításokat számológép használata nélkül és add meg az eredményt **hexadecimális** formában. Az alkalmazott számrendszerek jelölésére a feladatokban az órán is használt jelölésrendszer érvényes. A megoldások során maximum 8 biten ábrázolható számokat feltételezz. A műveletek működésénél a DYC működési elvét feltételezd. (2 pont).

$$\$3A + \%10011001 =$$

$$\text{SHL}(\$5C) =$$

$$\$A5 - 65 =$$

$$\text{SHR}(2 * \$93) =$$

- 2.) Mi lesz az **akkumulátor** tartalma a következő DIY Calculator Assembly kódrészlet megjelölt pontjain (4 pont)?

```
BLDX    $0001
BSTX    [TEMP16]
LDA     [TEMP16, X]    # %  _ _ _ _ _
SHL
ADD     126
JNN     [TOHERE]
DECA
TOHERE: DECA
        PUSHA        # %  _ _ _ _ _
LDA     $AE
JN      [TOHERE2]
INCA
TOHERE2: INCA        # %  _ _ _ _ _
        POPA
        SHR
        SHL
        STA     [$F031]    # %  _ _ _ _ _
```

- 3.) Írj olyan DIY Calculator Assembly programot, melyben bekérsz a billentyűkről (\$F011) két hexadecimális számot 0 és 9 között, majd a képernyőre irod az összes decimális számjegyet a két beolvasott között, beleértve a beolvasott két számot is. Előre nem tudod, melyik szám lesz a nagyobb. (6p)

Példa:

Be: 8, 4

Ki: 4 5 6 7 8

- 4.) Írj olyan DIY Calculator Assembly programot, mely az enter gomb lenyomásáig billentyűkódot olvas a workbench-hez tartozó billentyűzetről (\$F008), majd az enter (\$13) beolvasása után kiírja a beolvasott karaktersorozatot visszafelé a beolvasás sorrendjében (\$F031). (5p)

5.) Írj olyan DIY Calculator Assembly **szubrutint**, mely induláskor a veremben egy nyolcbites értéket vár. A szubrutin cserélje ki a kapott érték első négy bitjét az utolsó négygel, s ezután rakja vissza a verembe, majd térjen vissza. (7p)

Példa:

Híváskor a verem tetején: \$F5 (vagy: %11110101)

Visszatérés után a verem tetején: \$5F (vagy: %01011111)

6.) Írj olyan DIY Calculator Assembly programot, mely folyamatosan egy hexadecimális karaktert (0-F) ismételtet a kijelzőn (\$F031). A program végig ellenőrizze érkezik-e megszakítás és ha igen cserélje le a megjelenítendőt a nála eggyel nagyobb értékűre -- periodikus határfeltétellel. (0->1, 1->2, 9->A, **F->0**) Vigyázz, hogy a megszakítás kezelésekor az ACC és a státuszregiszterek értéke ne sérüljön (azaz a megszakításkezelő szubrutin végén álljon vissza az eredeti állapotába). (6p)

Példa futás:

CCCCCCCDDDDDEEEEEFFFFFFFFF000000

 ↑ ↑ ↑ ↑
 m e g s z a k í t á s o k