

ΓΡΑΠΤΗ ΕΞΕΤΑΣΗ ΣΤΟ ΜΑΘΗΜΑ "Συστήματα Μικροϋπολογιστών"

(ΘΕΜΑ 3^ο – ΣΥΝΟΛΟ 2 Μονάδες)

Έναρξη 13:40' - ΔΙΑΡΚΕΙΑ 30' + 10' Παράδοση: 14:20'

ΟΝΟΜΑΤΕΠΩΝΥΜΟ: ΣΤΕΦΑΝΑΚΗΣ ΓΕΩΡΓΙΟΣ el18436

ΘΕΜΑ 3ο: (2 ΜΟΝΑΔΕΣ): Σε ένα προσωπικό υπολογιστή, να γραφεί πρόγραμμα σε Assembly με 80x86 που να δέχεται από το πληκτρολόγιο τέσσερις (4) δεκαδικούς αριθμούς (D_3, D_2, D_1, D_0 με τη σειρά αυτή) για να αποτελέσουν δυο μονοψήφιους και ένα διψήφιο δεκαδικό αριθμό και να κάνει τον εξής υπολογισμό: $P = (D_3 + D_2) \times (D_1 \times 10 + D_0)$. Το πρόγραμμα τυπώνει στην οθόνη τα μηνύματα εισόδου και τους εισαγόμενους αριθμούς. Όταν συμπληρωθούν 4 έγκυροι δεκαδικοί αριθμοί να αναμένει τον χαρακτήρα 'H' και μετά να τυπώνει το αποτέλεσμα σε δεκαεξαδική μορφή 3 ψηφίων αν είναι <500Hex, αλλιώς το μήνυμα overf, αυστηρά όπως φαίνεται παρακάτω:

GIVE 1ST NUMBER = 9
GIVE 2ND NUMBER = 7
GIVE 3ND NUMBER = 58
RESULT = 3A0 ή RESULT = overf

Να θεωρήσετε δεδομένες τις μακροεντολές (σελ. 361-2, 373) του βιβλίου και μπορείτε να κάνετε χρήση των ρουτινών DEC_KEYB και PRINT_HEX χωρίς να συμπεριλάβετε τον κώδικά τους. Για την διευκόλυνσή σας, δίνονται οι πρώτες εντολές που αποτελούν τον 'σκελετό' του ζητούμενου προγράμματος.

ΑΠΑΝΤΗΣΗ

INCLUDE MACROS

DATA_SEG SEGMENT

```
MSG1 DB 0AH,0DH, 'GIVE 1ST NUMBER= $'
MSG2 DB 0AH,0DH, 'GIVE 2ND NUMBER= $'
MSG3 DB 0AH,0DH, 'GIVE 3ND NUMBER= $'
MSG4 DB 0AH,0DH, 'RESULT = $'
MSG5 DB 0AH,0DH, 'overf' ; overflow message
DATA_SEG ENDS
```

CODE_SEG SEGMENT

ASSUME CS:CODE_SEG, DS:DATA_SEG

MAIN PROC FAR

```
MOV AX, DATA_SEG
MOV DS, AX
```

ADR1:

```
PRINT_STR MSG1
CALL DEC_KEYB
CMP AL, 'Q'
JE QUIT
MOV BL, AL
PRINT_STR MSG2
CALL DEC_KEYB
CMP AL, 'Q'
JE QUIT
ADD BL, AL ; D3 + D2
PRINT_STR MSG3
CALL DEC_KEYB
CMP AL, 'Q'
JE QUIT
```

```

MOV DL, 10
MUL DL      ; D1 * 10
MOV DL,AL
CALL DEC_KEYB
CMP AL,'Q'
JE QUIT
ADD AL,DL   ; D1 * 10 + D0
MUL BL      ; praksi P
MOV CX, 3
CMP 500H, AX
JL OVERFLOW

H_PRESS:
CALL DEC_KEYB      ; wait until H is pressed
CMP AL, 'H'
JE ADR2
JMP H_PRESS

ADR2:
ROL AX,1
ROL AX,1
ROL AX,1
ROL AX,1
MOV DL,AL
AND DL,0FH
CMP 500H, AX      ; is result >= 500H
JG OVERFLOW
PUSH AX
CALL PRINT_HEX
POP AX
LOOP ADR2
JMP ADR1

OVERFLOW:
PRINT_STR MSG5   ; print overf
JMP ADR1

QUIT: EXIT
MAIN ENDP
CODE_SEG ENDS
END MAIN

```