# COMPUTER ARCHITECTURE

Written exam, January 26, 2012, gr 912

I.

The structure and the role of the EXE header. What is it, which is its purpose, where is it located, which specific information does it provide and to whom, why is it important and when exactly is it involved during the lifetime of a program. Explain in this context the concept of "relocatable item", how the relocation mechanism works and why this mechanism is necessary. Which are the steps of loading an EXE program in memory and which are the registers initializing values?

II.

A string of double words is given. It is required to build and print the string of byte ranks that have the maximum value from each doubleword (considering them unsigned). Also it is required to obtain and print on the screen the sum of these bytes (considering them this time as being signed). Explain the algorithm, justify and comment accordingly the source code. When explaining, focus on problematic and difficult aspects involved in the given solution.

Example: dd 1234A678h , 12345678 9h , 1AC3B47Dh, FEDC9876h … the corresponding string of bytes ranks being "3421"

III.

Which will be the contents of the AX and the DX registers after running the sequence:

a)
```
a  db  3
b  dd  256
c  dw 256
mov  cx,  word  ptr  b + 1
xchg  ch,  cl
neg  ch
not  cl
add  cx,  1
les di,  b + 2
push  ds
pop  es
mov  ah,  es : [di]
mov  bx,  ex : [di]
```

b)
```
mov  ah,  128
neg  ah
xor  al,  al
cwd
cbw
idiv  dx
```

Explain and justify the effect of every source line (in the case that you consider it syntactically correct, of course – if not explain why is it incorrect and ignore further that line …) , showing what exactly does that particular instruction and which will be the contents of the involved registers, in each of the 2, 10, 16 numeration bases (signed and unsigned interpretations).

IV.

a) The following 2 instructions sequence is given:
```
mov ax,  bx
add ax,  di
…
```
Write one single asm instruction which has the same effect as these 2 instructions. Explain the reason for which the effect is the same.

b) Present XLAT instruction: syntax, semantics, adequate source code example for putting into evidence its usefulness. Is it necessary of not at the level of the 80x86 assembly language?

I – 2.5 pts, II – 3 pts, III – 2 pts, IV – 1.5 pts