**Operatori de tip si tipuri de date asociate operanzilor**

(materialul de mai jos este cel discutat impreuna cu adnotarile si comentariile dezvoltate in comun)

v d? ….

a d?...

b d?...

Push v - ok ! stack 🡨offset v (L-value := R-value !!)

Push [v] - Operation size not specified !!

Push dword [v] – ok !

Push word [v] – ok + warning !

Push byte [v] – NEVER EVER !!! Syntax error !

Push [eax] - Operation size not specified !!

Push dword [eax] – ok !

Push word [eax] – ok + warning !

Push 15 ; - PUSH DWORD 15 !!!!

Pop [v] ; Operation size not specified !!

Pop word/dword [v]; ok !

Pop v ; syntax error ! - v:=top\_of(stack); 2:=3 !!!

Pop dword b ; - syntax error !

Pop [eax] ; Operation size not specified !!

Pop word/dword [eax]; ok !

Pop 15 ; 15 is a R-value !!

Pop [DS:15] ; Operation size not specified !!

Pop word/dword [15] – OK sintactic !!!! – DAR… cel mai probabil veti obtine “Memory violation error!!” (run-time error)

Mov [v],0 ; syntax error !

Mov byte/word/dword [v],0 – ok !!!

Mov [v], byte/word/dword 0 – ok !!!

Div [v] ;

Imul [v+2] ;

Mov a,b ; a = R-value !!

Mov [a], b ; Op.size not specified !

Mov word/dword [a], b – OK !!!!

Mov byte [a], b – SYNTAX ERROR !!!! (similara cu mov ah,v)

Mov a,[b] ; a – R-value !!!

Mov [a], [b] ; - NU putem avea in L.a. amandoi operanzii din memorie !!!

Mul v – MUL reg/mem – syntax error !!!

Mul [v] – syntax error ! op size not specified !!!

Byte/word/dword

Mul eax ; ok !

Mul [eax] ; syntax error ! op size not specified !!!

Byte/word/dword

Mul 15 ; syntax error !

Pop byte [v] -

Pop qword [v] –

**Operatori de tip si tipuri de date asociate operanzilor**

**(material pregatit de mine in avans ca exemple + comentarii); Le-am pastrat pe amandoua desi sunt identice ca exemple pt a le studia comparativ**

v d? ….

a d?...

b d?...

Push v - ok, stack 🡨offset v (pe 32 biti)

Push [v] - syntax error – Operation size not specified !

Push dword [v] – ok

Push word [v] - ok

Push [eax] – syntax error…. Operation size not specified !

Push byte [eax] – syntax error….

Push word [eax] – ok

Push 15 ; PUSH DWORD 15

Pop [v] ; Op size not specified

Pop word/dword [v]

Pop v ; sintaxa este POP destinatie; destinatie must be a L-value !!!!!

…dar v este R-value !!!! ; acest pop v este similar ca operatie cu 2:=3 !!!

(Invalid combination of opcode and operands)

Pop dword b ; syntax error !

Pop [eax] ; Op size not specified

Pop (d)word [eax] ; ok!

Pop 15 ; 15 is NOT a L-value !! – syntax error

Pop [15] ; syntax error - Op size not specified

Pop dword [15] ; syntactic ok , cel mai probabil run-time error deoarece probabil [DS:15] va provoca Access violation !!

Mov [v],0 ; syntax error - Op size not specified

Mov byte [v], 0 ; ok !

Mov [v], byte 0

Div [v] ; op. size ?

Div word [v]; ok !

Imul [v+2] ; op. size ?

Imul word [v+2]; DX:AX = AX\*word de la adresa v+2

Mov a,b ; syntax error pt ca a NU este L-value, ci R-value fiind un offset determinabil ca si constanta la momentul asamblarii

Mov [a], b ; syntax error – op. size not specified !

Mov byte [a], b ; syntax error – OBJ file 16 sau 32….

Mov word [a], b ; ok !! – 2 octeti inferiori din valoarea offset-ului lui b !

Mov dword [a], b ; full offset 32 bits

Mov a,[b] ; Invalid comb. Of opcode and operands (a = R-value)

Mov [a], [b] ; Invalid comb. Of opcode and operands (NU putem avea 2 operanzi simultan din memorie)

Mul v – syntax error – MUL reg/mem

Mul word v - syntax error – MUL reg/mem

Mul [v] - op. size not specified

Mul dword [v]; ok !

Mul eax ; ok !

Mul [eax] ; op. size not specified

Mul byte [eax] ; ok !!!

Mul 15 ; Invalid comb. Of opcode and operands - – MUL reg/mem

Pop byte [v] – Invalid combination of opcode and operands

Pop qword [v[] – Instruction not supported in 32 bit mode !

**Clasificarea erorilor in informatica**

* **Eroare de sintaxa – ea este diagnosticata de asamblor/compilator !**
* **Run-time error (eroare la executie) – programul “crapa” – programul se opreste = program crash !!**
* **Eroare logica = programul functioneaza pana la capat sau ramane blocat in ciclu infinit, insa GRESIT dpdv LOGIC obtinand cu totul alte rezultate decat cele asteptate…**
* **Fatal: Linking Error !!! (de ex in cazul unei definitii dublate de variabila… 17 module si o variabila trebuie sa fie DEFINITA DOAR intr-un singur modul ! Daca ea este definite in 2 sau mai multe module se va obtine Fatal: Linking Error !!! – Duplicate definition for symbol A1 !!!)**

**The steps followed by a program from source code to run-time:**

* Syntactic checking (done by assembler/compiler/interpreter)
* OBJ files are generated by the assembler/compiler
* Linking phase (performed by a LINKER = a tool provided by the OS, which checks the possible DEPENDENCIES between this OBJ files/modules); The result 🡪 .EXE file !!!
* You (the user) are activating your exe file by clicking or enter-ing…
* The LOADER of the OS is looking for the required RAM memory space for your EXE file. When finding it, it loads the EXE file AND performs ADDRESS RELOCATION !!!!
* In the end the loader gives control to the processor by specifying THE PROGRAM’s ENTRY POINT (ex: the start label) !!! The run-time phase begins NOW…

Mark Zbirkowski – semnatura EXE = ‘MZ’