**Laborator 1**  
  
**1. Utilizarea programelor tasm, tlink, td**  
  
**tasm** - este un asamblor, traduce un program scris in limbaj de asamblare in format obiect (fisier.asm -> fisier.obj)  
- lansat fara parametri furnizeaza sintaxa corecta  
- exemplu de utilizare : tasm exemplu sau tasm exemplu.asm  
  
**tlink** - editor de legaturi transforma un fisier din format obj in format executabil. Poate sa grupeze mai multe fisiere obj intr-un singur program executabil.  
- fis1.obj + fis2.obj + ... -> fisier.exe (sau fisier.com)  
- lansat fara parametri furnizeaza sintaxa corecta  
- ex. de utilizare : tlink exemplu sau tlink exemplu.obj  
- optiuni mai importante: /v, /t  
  
Odata obtinut programul executabil, acesta se poate lansa in executie.  
  
**td** - este un depanator, un program care permite rularea pas cu pas a programelor executabile.  
- ex. de utilizare td exemplu sau td exemplu.exe  
- TD permite doua tipuri de depanare:  
1. Depanare la nivel de limbaj masina - aceasta este posibila pentru orice program executabil, exe sau com.  
2. Depanare la nivel de limbaj sursa - posibila pentru acele programe executabile care includ informatii de depanare si pentru care este prezent fisierul sursa.  
  
**OBSERVATIE**  
  
Pentru depanare la nivel de limbaj masina vom folosi:  
...> tasm exemplu  
...> tlink exemplu  
...> td exemplu  
  
Pentru depanare la nivel de limbaj sursa vom folosi:  
...> tasm /zi exemplu  
...> tlink /v exemplu  
...> td exemplu  
  
Urmarirea programului cu Turbo Debugger se va face in fereastra CPU (se deschide din meniul View|CPU). Aici avem continutul registrilor si flagurilor procesorului si programul ce se executa.Acesta este prezentat atat in limbaj de asamblare cat si direct in cod masina (adica, octet cu octet, continutul memoriei), continutul segmentului de date (in hexa) si continutul stivei. Putem pozitiona zona vizibila din program, zona de date si stiva folosind comanda Go to din meniul local al ferestrei (meniul activat cu Alt-F10). De asemenea, putem afla sau modifica valoarea unei variabile, folosind fereastra de dialog Evaluate/Modify (deschisa cu Ctrl-F4)

**2. Un exemplu minimal de program in limbaj de asamblare**  
  
ASSUME cs: code, ds:data  
   ;spunem asamblorului care sunt segmentele folosite de noi  
data SEGMENT   
   ;data - segmentul de date in care vom defini variabilele  
data ENDS  
code SEGMENT  
   ;code - numele segmentului de cod  
start:  
   mov ax,data    ;adresa segmentului de date se copiaza in ax  
   mov ds,ax    ;continutul lui ax se copiaza in ds  
   ; .......  
   ;aici avem instructiunile programului nostru  
   ;........  
   mov ax,4C00h  
   int 21h   ;finalul executiei programului  
code ENDS  
END start