

# Curs 13

## Implementarea apelului de subprogram în informatică

- etapele utilizării unei funcții:
  - o apelul funcției
  - o intrare
  - o ieșire
- cdecl, stdcall
- resurse volatile: eax, ecx, edx, eflags
- Sarcini:
  - 1) Salvare resurse volatile în uz: push registru
  - 2) Pregătire argumente (stivă, conform cerinței): push
  - 3) Efectuare apel: call
    - call subrutină când subtrina este linkeditată static
    - call [subrutină] dacă este dinamică (la link time)
    - call registru sau call [variabilă] pt dinamică la runtime
- Cod de intrare:
  - a) Configurare cadru de stivă
  - b) Rezervare spațiu eventuale variabile locale
  - c) Salvare resurse nevolatile (dacă este cazul)
- Rezumarea unui nou cadru de stivă:

push ebp

mov ebp, esp

...

mov esp, ebp

pop ebp

- Cod ieșire:

- a) Restaurare resurse volatile
- b) Eliberare variabile locale
- c) Dezafectare cadru stivă

mov esp, ebp

pop ebp

Apelant	Apelat	Cod apel	Cod intrare	Cod ieșire
C	C	Compilatorul C	Compilatorul C	Compilatorul C
C	Asm	Compilatorul C	Programator asm	Programator asm
Asm	C	Programatul asm	Compilator C	Compilator C

Asm	Asm	Call (cu salvare adresă de revenire)	Nimic obligatoriu (totul cum vrei)	Ret(recuperare adresă de revenire)
-----	-----	--------------------------------------	------------------------------------	------------------------------------