

Programare in limbaj de asamblare

Conf. dr. ing. Anca Hangan

Anca.Hangan@cs.utcluj.ro

<http://users.utcluj.ro/~ancapop>

program.c

...

int i;

...

scanf("%d",&i);

...

```
program.c
```

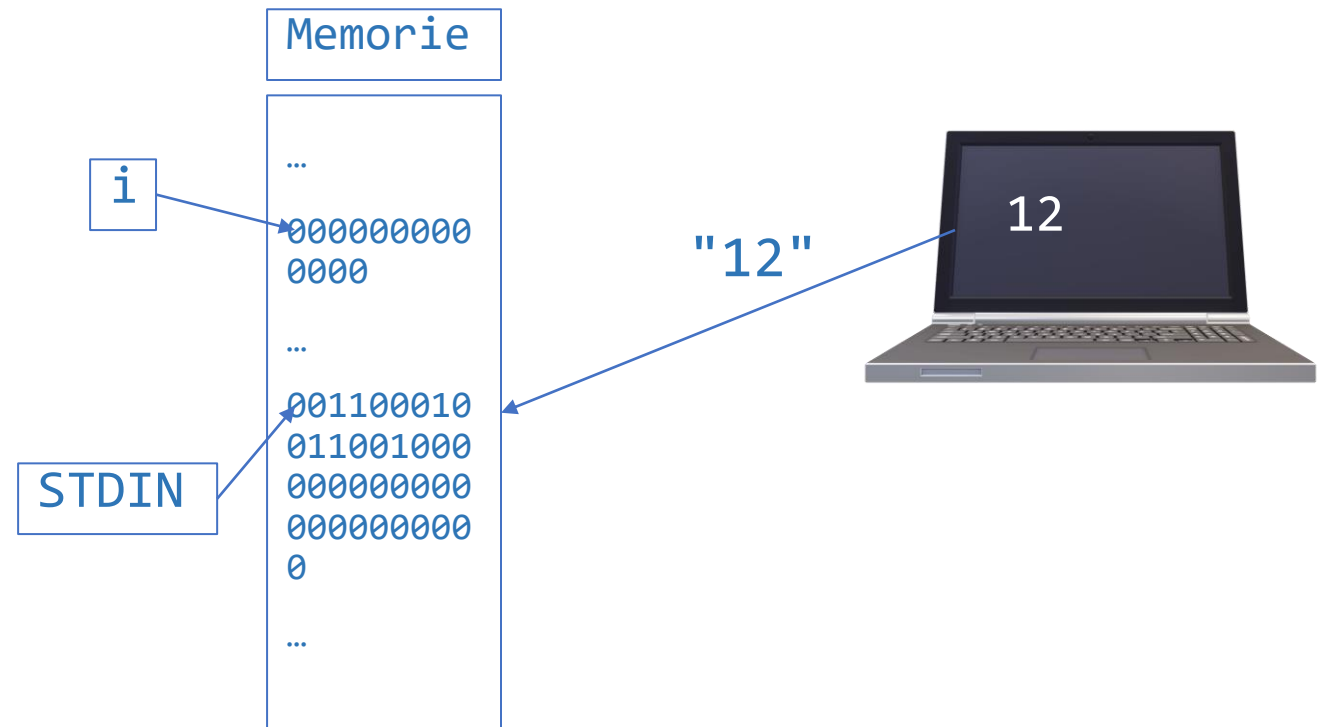
```
...
```

```
int i;
```

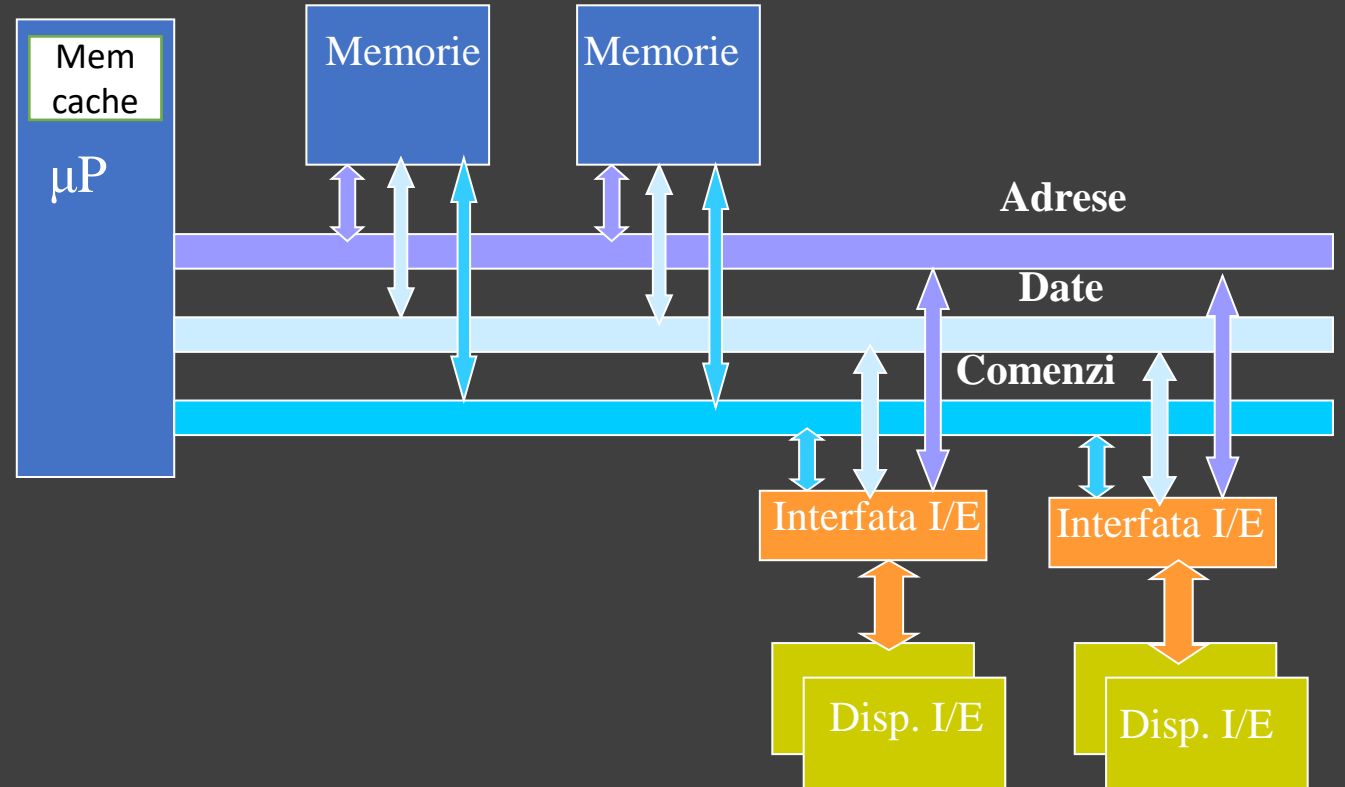
```
...
```

```
scanf("%d",&i);
```

```
...
```



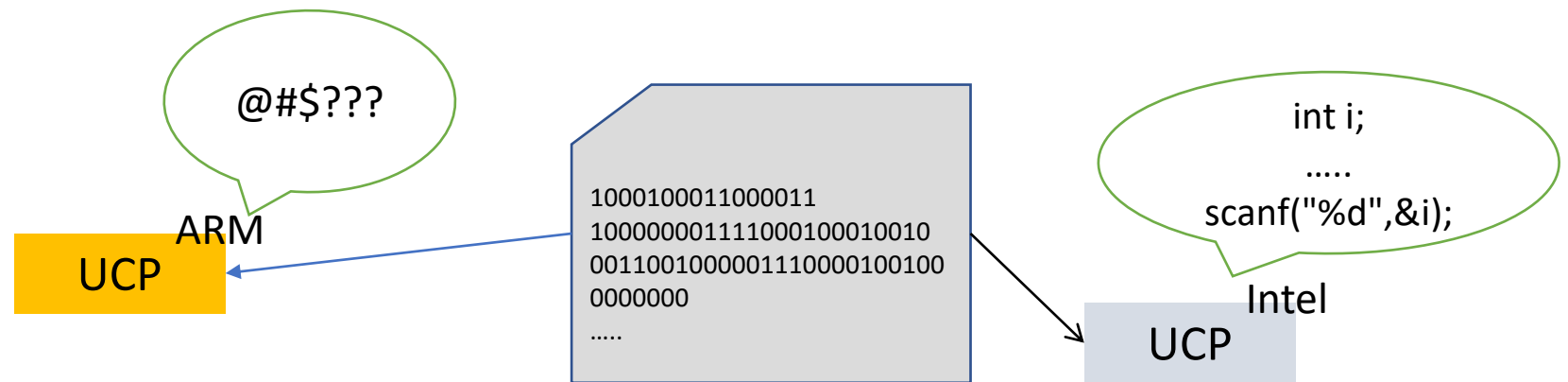
Structura unui sistem de calcul



Limbajul de programare

Cod masina

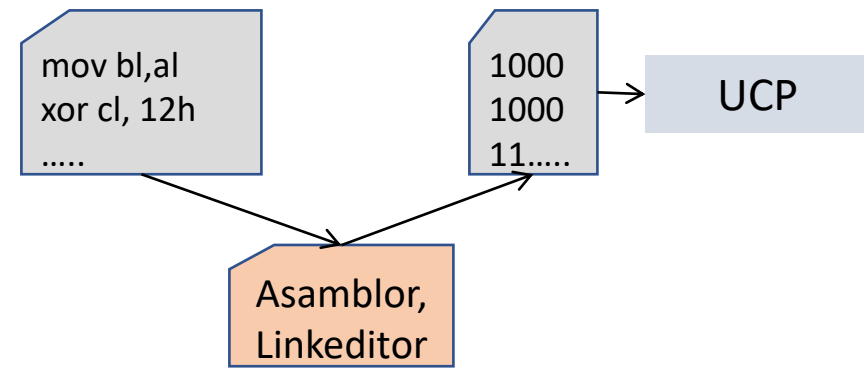
- Instructiuni reprezentate in cod binar
- Fiecare tip de procesor are un format specific de reprezentare



Limbajul de programare

Asamblare

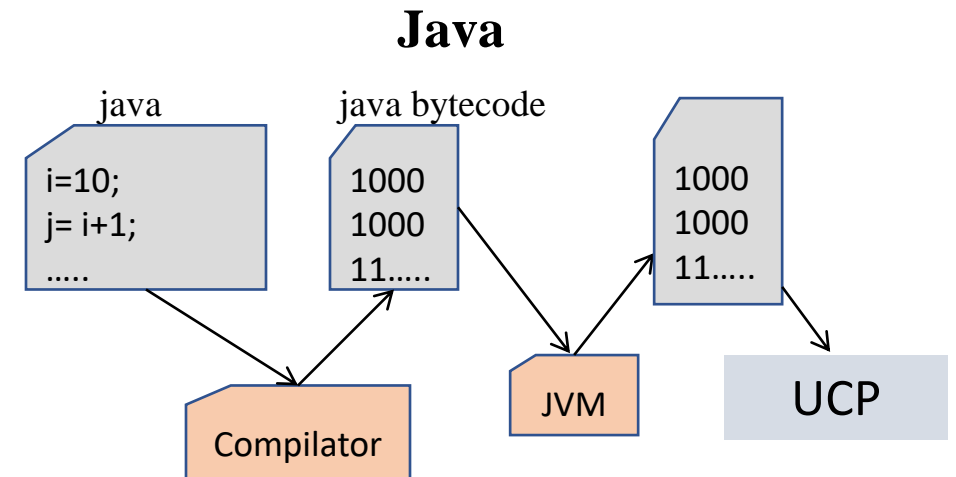
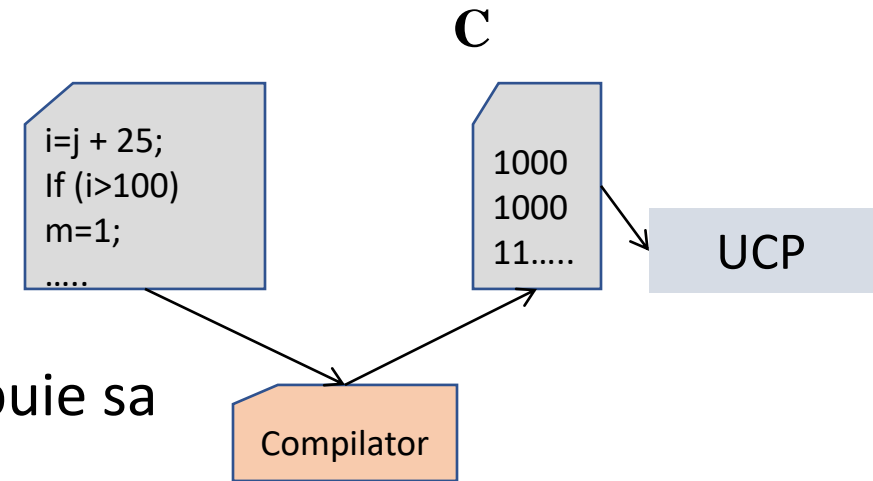
- Forma simbolica a codului masina
- Simboluri si cuvinte inteligibile pentru om
- Instructiuni direct translatabile in cod masina
- Opereaza cu resursele fizice ale unui calculator: registre, memorie, interfete I/O, etc.



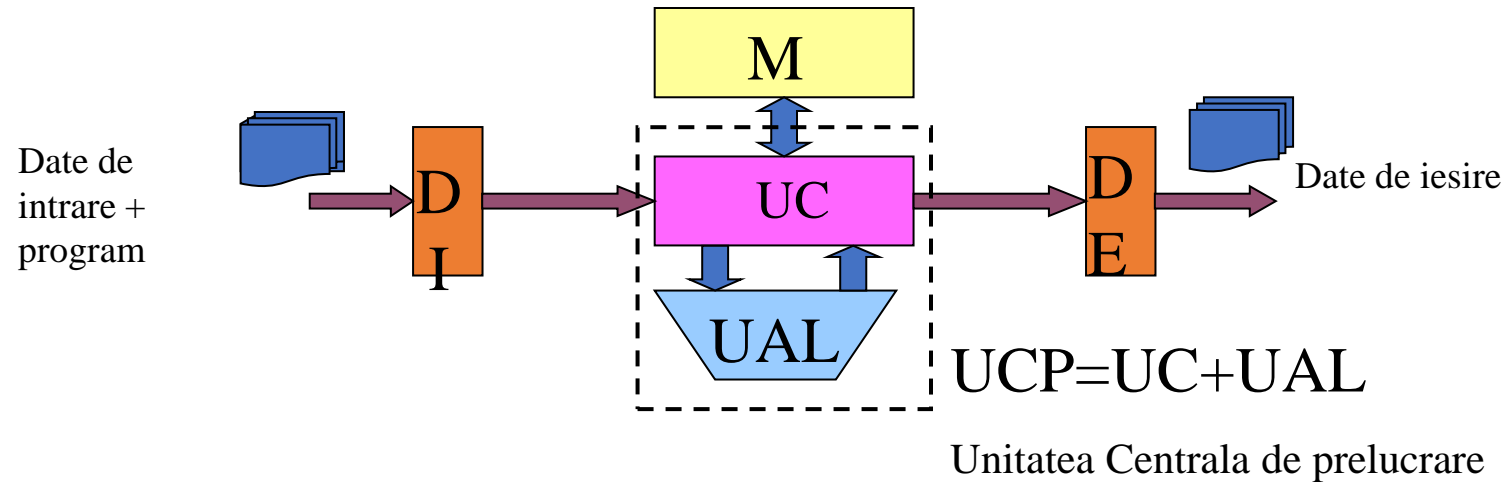
Limbajul de programare

- **De nivel inalt**

- Mai usor de inteles
- Mai putine linii de cod
- Portabile (daca treci pe alt procesor, nu trebuie sa rescrii programul)



Modelul unui sistem de calcul (von Neumann)



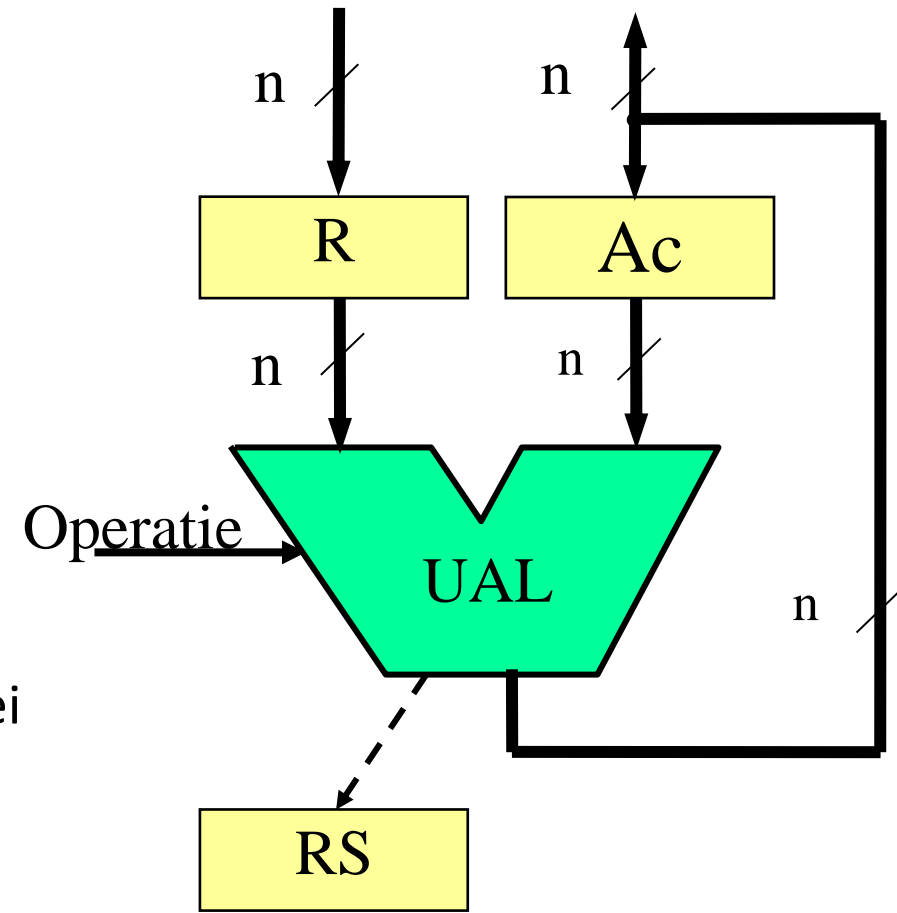
- Unitatea de control (UC)
- Unitatea aritmetico-logica (UAL)
- Registre (R)
- Memoria (M)
- Dispozitiv de intrare (DI)
- Dispozitiv de iesire (DE)

Unitatea de comanda

- Circuit secvential (automat de stare)
- Responsabilitati
 - Citirea instructiunilor
 - Decodificarea instructiunilor
 - Generarea de semnale pentru executia instructiunilor
- Componente
 - Generatorul de tact (ceas)
 - Generatorul de faze (secventierea executiei unei instructiuni)
 - Unitatea de decodificare instructiuni
 - Blocul de comanda si control (semnale pentru celelalte componente ale UCP)

Unitatea aritmetico-logica

- Efectueaza operatii aritmetice si logice: +,-,/,*,SI, SAU, NU,XOR
- Deplasari/rotiri
- Este (in general) un circuit combinational
- Ac – registru acumulator
 - Pastreaza un operand si rezultatul operatiei
- R – registru pentru al doilea operand
- RS registru de stare



Registre

- Pastrarea temporara a datelor
- **Registre generale** – utilizate pentru operatii aritmetice si logice
- **Registre speciale** – utilizate pentru:
 - Pastrarea adresei instructiunii urmatoare (numarator de program)
 - Pastrarea instructiunii curente (registru de instructiuni)
 - Pastrarea starii programului (registru de stare)
 - Adresare
 - Testare, control, etc.

Executia unei instructiuni

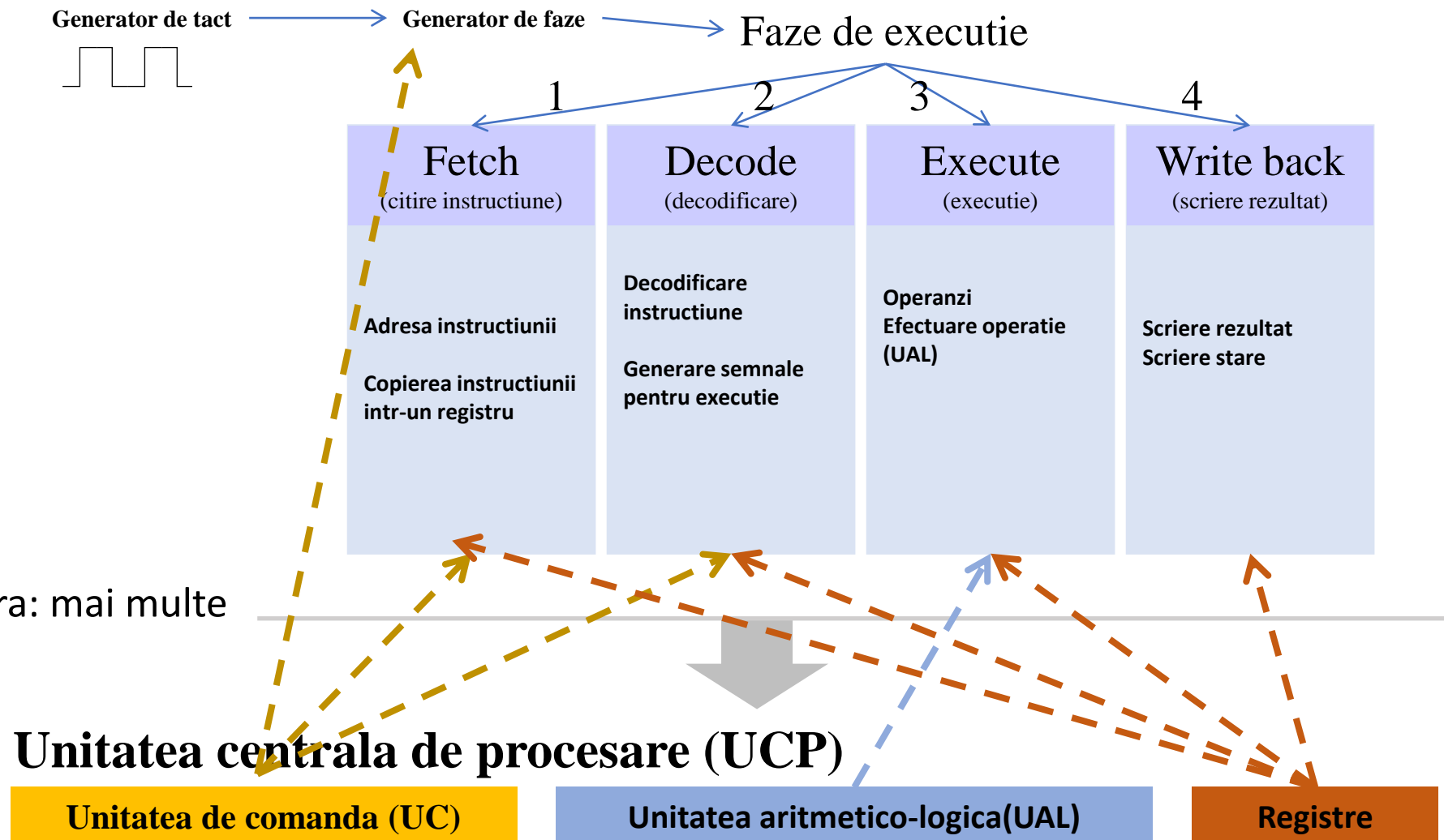
- Intern UCP->faze de executie

- Pe magistrala sistemului->

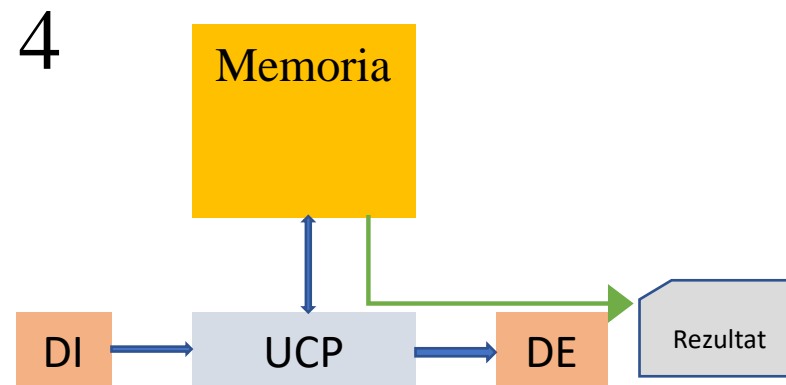
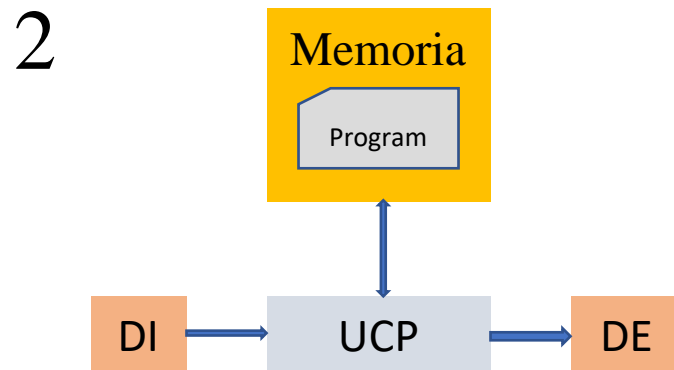
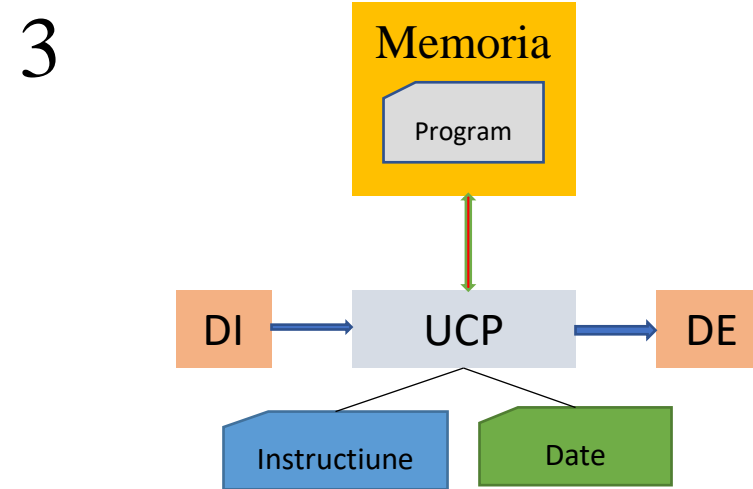
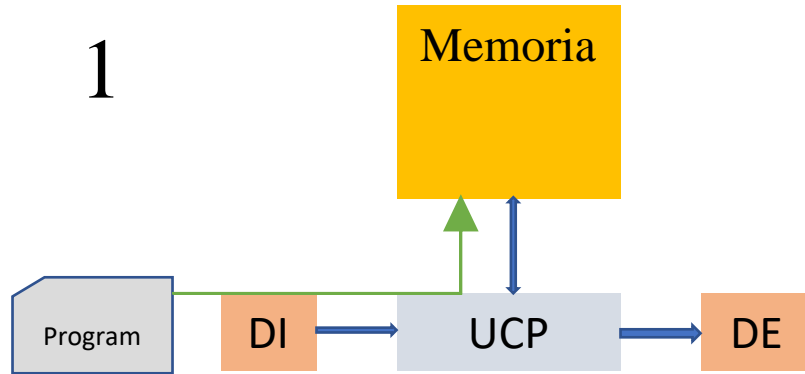
Operatii cu memoria:

- Citire instructiune
- Citire operand
- Scriere rezultat

- Executie pipeline/superscalară: mai multe instructiuni in acelasi timp



Executia unui program



De ce sa inveti limbajul de asamblare?

- Vrei sa intelegi modul in care lucreaza un calculator
- Vrei sa ai access direct la resursele unui calculator
- Vrei sa scrii programe eficiente (spatiu ocupat, timp de executie)
- Vrei sa programezi sisteme incapsulate
- Vrei sa intelegi modul in care functioneaza un program si nu ai codul sursa, doar fisierul executabil
- Vrei sa modifichi un program, dar ai doar fisierul executabil

Se considera cunoscut

- Circuite secventiale si combinational
- Reprezentarea numerelor in diferite baze de numeratie (2,8,16)
- Conversii dintr-o baza in alta
- Operatii cu numere in baza 2 si 16
- Reprezentarea numerelor cu semn in C2
- Formate binare de reprezentare BYTE, WORD, DOUBLE-WORD
- Reprezentarea in virgula fixa si virgula mobila
- Codificarea ASCII, BCD
- Conventii de stocare si transmite a datelor: little endian, big endian

Formate binare de reprezentare

- Bit:
 - binary digit;
 - unitatea elementara de informatie
 - starea unui bistabil, sau a unei celule elementare de memorie
- Octet (byte):
 - grup de 8 biti
 - unitatea elementara de adresare la cele mai multe calculatoare actuale (inclusiv Intel x86)
 - poate reprezenta: o valoare numerica, **un caracter (cod ASCII)**, un set de variabile (semnale) logice

Formate binare de reprezentare

- Octet (continuare)

7	6	5	4	3	2	1	0
---	---	---	---	---	---	---	---

D7 - bitul cel mai semnificativ

D0 - bitul cel mai puțin semnificativ

- Cuvant (word)

15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
----	----	----	----	----	----	---	---	---	---	---	---	---	---	---	---

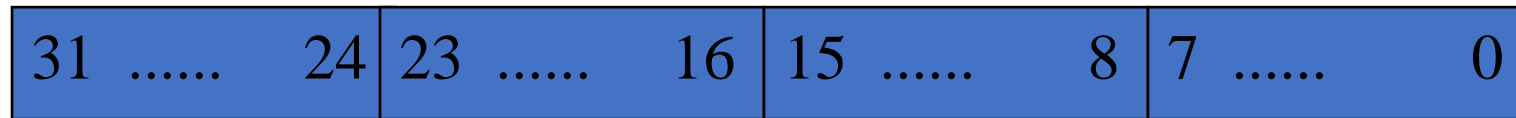
D15-D8 - octetul superior (High)

D7-D0 - octetul inferior (Low)

- folosit pt. reprezentarea întregilor, simplă precizie

Formate binare de reprezentare

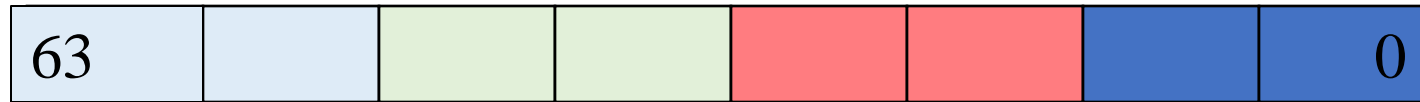
- Dublu-cuvant (double word):



- 32 biti, 4 octeti, 2 cuvinte
- D31-D16 cuvantul superior
- D15-D0 cuvantul inferior
- folosit pentru reprezentarea numerelor in virgula fixa (dubla precizie la sistemele pe 16 biti) sau in virgula flotanta

Formate binare de reprezentare

- Cuadruplu-cuvant (quad-word)



- 64 de biti, 8 octeti, 4 cuvinte, 2 dublucuvinte
- folosit pentru reprezentarea numerelor in virgula fixa (dubla precizie pt. sist. pe 32 biti) si in virgula flotanta
- Formate extinse:
 - 80 de biti - pt. reprezentarea numerelor in virgula flotanta (formate interne/intermediare)

Conventii de stocare/transmitere a datelor

- Little-endian/ Big-endian

- Little-endian: partea mai putin semnificativa la adresa mai mica

- Octeti: 3AH, 33H, 12H



Adresa: x x+1 x+2 x+3 x+4

- Cuvinte: 1234H, 56ABH, FFFFH



Adresa: x x+1 x+2 x+3 x+4 x+5 x+6 x+7

- Dublu-cuvante: 01234567H, 89ABCDEFH



Adresa: x x+1 x+2 x+3 x+4 x+5 x+6 x+7 x+8

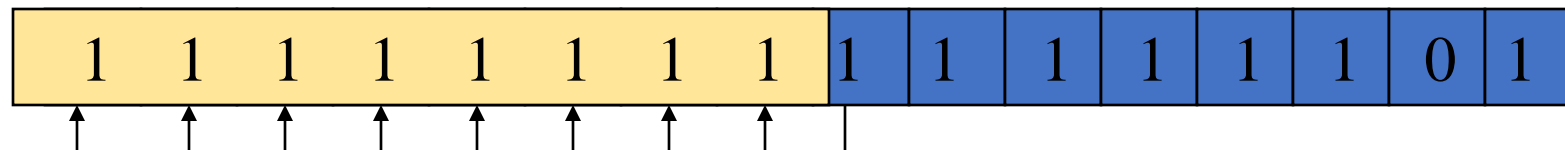
Aritmetica in complement fata de 2

7+	0000.0111+	7+
-10	1111.0110	246
<hr/>	<hr/>	<hr/>
-3	1111.1101	253

- in C2 operatiile aritmetice sunt identice cu aritmetica numerelor pozitive ->

ESTE O CHESTIUNE DE INTERPRETARE !!!

- la reprezentarea numerelor negative trebuie sa se precizeze lungimea de reprezentare
- pt. cresterea lungimii de reprezentare: **Extensia de semn**



Reprezentarea in virgula flotanta

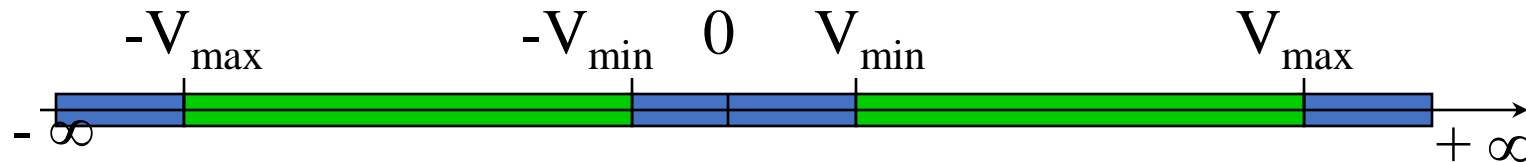
- scopul: reprezentarea numerelor foarte mari si foarte mici
- forma de reprezentare: semn, caracteristica si mantisa



- Simpla precizie: 32 de biti
 - 1 Semn, 8 Caracteristica, 23 Mantisa
- Dubla precizie: 64 de biti
 - 1 Semn, 11 Caracteristica, 52 Mantisa
- Precizie extinsa: 80 de biti
 - 1S,15C,1I,63M
- caracteristica = $\text{exponent} + 1/2 (\text{domeniu_exponent})$

Reprezentarea in virgula flotanta

- limite de reprezentare:



- unde: $V_{\max} = 2^{(2^{(c-1)}-1)*0,11111..1}$
 $V_{\min} = 2^{-(2^{(c-1)}-1)*0,1}$
- rezolutia (granularitatea) absoluta \rightarrow variabila
- rezolutia (granularitatea) relativa \rightarrow constanta
- reprezentarea in flotant este **discreta**,
NU modeleaza pe deplin numerele reale

Codificarea alfanumerica

- Standardul ASCII:
 - se codifica: litere, cifre, semne de punctuatie, semne grafice, comenzi de formatare, comenzi de control al transmisiei
 - se folosesc 7 biti/cod (+1 bit paritate) sau 8 biti/cod (pt. standardul extins)
- Exemple:
 - litere mari: A- 41H, B- 42H, C - 43H
 - litere mici: a - 61H, b - 62H, c - 63H, ..
 - cifre: 0 - 30H, 1 - 31H, 9 - 39H
 - altele: spatiu - 20H, CR - 0DH, LF - 0AH.....

Continutul cursului

- Programarea in LA - procesoare de la Intel cu arhitectura pe 32 de biti (IA-32)
- Scrierea unui program
- Arhitecturi Intel
- Moduri de operare ale procesorului IA-32
- Organizarea memoriei la arhitecturile IA-32
- Mediul de executie IA-32
- Registri accesibili din program
- Adresarea operanzilor

Continutul cursului

- Tipuri de date fundamentale
- Instructiuni generale
- Reutilizarea codului: macro si proceduri
- Instructiunile unitatii in virgula mobila x87
- Instructiuni MMX
- Intreruperi
- Alte arhitecturi si seturi de instructiuni

Program in asamblare

```
.386
.model flat,stdcall;

includelib msvcrt.lib
extern exit:proc
public start

.data

;aici declaram date

.code
start:

;aici se scrie codul

push 0
call exit
end start
```

Bibliografie

- Manuale Intel IA-32
- Referinta instructiuni
- Lucrarile practice (Moodle)
- Cursuri: MSTeams (**v172kdf**) teste pe Moodle
- Tools (le gasiti pe Moodle):
 - Asamblor + linkeditor de la Microsoft: MASM 32 bit
 - Depanare programe: OllyDbg
 - Mediu de programare: Notepad++ cu plugin MASM
- Comunicare: e-mail, MSTeams

Conditii de promovare

- 100% prezenta la laborator
- Nota laborator minim 5
 - Lucrari laborator – teste pe Moodle
 - Teme
 - Colocviu
- Nota examen minim 5 ($4.99 = 4$)
- Nota finala = $0.3 * \text{nota lab} + 0.7 * \text{nota examen}$
- **Nu se dau puncte din oficiu**
- Puncte din oficiu – teste pe Moodle, prezenta, participare active la curs

Exercitii

- Reprezentati in C2 numerele: 22, -22
- Reprezentati in VM format scurt (32 de biti): 22.43