



Facultatea de Inginerie Electrică



UNIVERSITATEA TEHNICĂ
DIN CLUJ-NAPOCA



PCLP 2

Programarea calculatoarelor si limbaje de programare 2

PCLP2

An I semestrul II



"Coding is easy when you C it in action."

Cap. 3

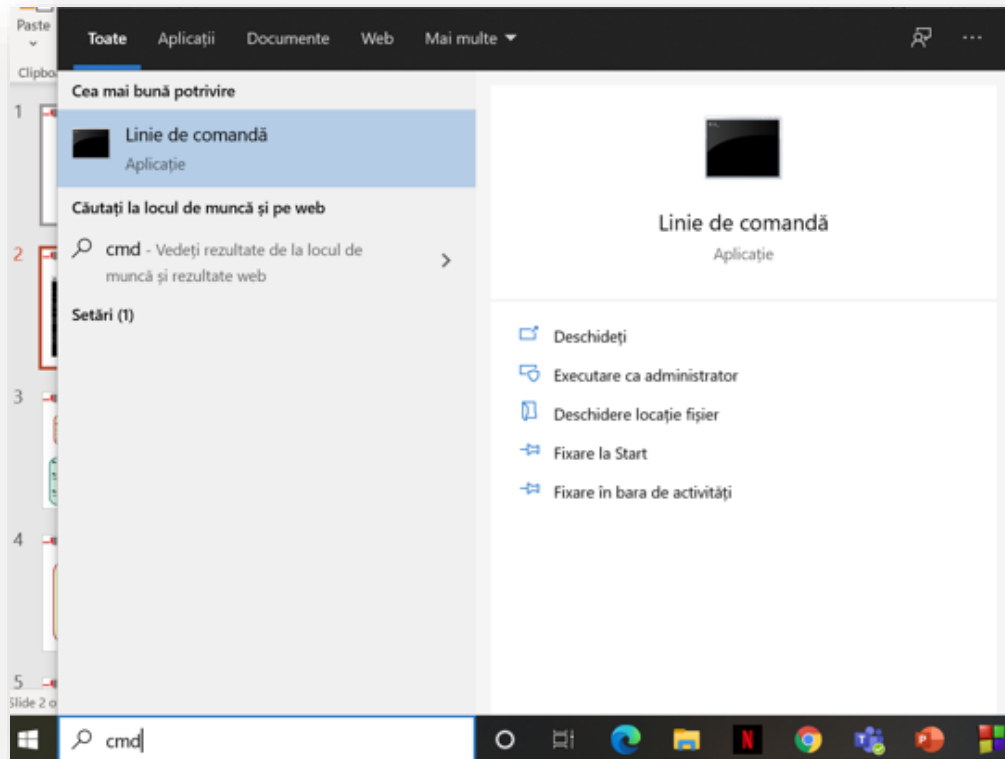
Linia de comanda

3.1. Argumentele liniei de comanda

3.2. Headerul functiei main(). Tipul returnat de functia main(). Parametri functiei main()

Cap 3: Linia de comanda

Ce este linia de comanda?



```
Linie de comandă
16/03/2021 10:37 <DIR>      obj
                    3 File(s)      1,516 bytes
                    4 Dir(s)  69,246,136,320 bytes free

C:\Laura\PCLP2\nume>cd bin

C:\Laura\PCLP2\nume\bin>cd debug

C:\Laura\PCLP2\nume\bin\Debug>dir
Volume in drive C is OS
Volume Serial Number is 24DE-DA3B

Directory of C:\Laura\PCLP2\nume\bin\Debug

16/03/2021 10:38 <DIR>      .
16/03/2021 10:38 <DIR>      ..
16/03/2021 10:38                54,716 nume.exe
                    1 File(s)      54,716 bytes
                    2 Dir(s)  69,246,201,856 bytes free

C:\Laura\PCLP2\nume\bin\Debug>nume
Salut (null) !
C:\Laura\PCLP2\nume\bin\Debug>nume Nicu
Salut Nicu !
C:\Laura\PCLP2\nume\bin\Debug>
```

Cap 3: Linia de comanda

3.1. Argumentele liniei de comanda

DEFINIRE

Linia de comanda este compusa din **numele programului** ce urmeaza a fi executat si **argumentele liniei de comanda**

Argumentele liniei de comanda sunt informatiile ce urmeaza numelui de program ce urmeaza a fi executat. **Argumentele sunt despartite prin spatii** .

Ex. 1: Programul suma.exe aduna 2 nr. specificate in linia de comanda

```
> suma 12 256
```

Argumentele liniei de comanda

Ex. 2: Programul invers.exe afiseaza toate cifrele din componenta numarului specificat in ordine descrescatoare pina la 0 (cronometru)

```
>nrinv 20
```

Ex. 3: Programul prog.exe utilizeaza ca date de intrare fisierul int.txt iar rezultatele le tipareste in out.txt

```
>prog.exe <int.txt >out.txt
```

Linia de comanda



TEST

1. Functia main() are parametri ?
2. Ce returneaza functia **main()** ?
3. Care forma este corecta?

a) void main(void)
{ ... }

b) main(void)
{ ... }

c) int main()
{ ... }

d) int main(int argc, char *argv[]) {...}

Raspuns corect

1. da
2. int
3. c,d

Cap 3: Linia de comanda

3.2. Argumentele functiei main()

DEFINIRE

Headerul functiei **main()**

```
int main(int argc, char *argv[])  
{ ...return 0 }
```

argc este de tip intreg si memoreaza numarul de argumente din linia de comanda. Are valoarea 1 implicit (cel putin numele programului este dat in linia de comanda).

argv este un tablou de pointeri catre siruri de caractere . Fiecare element al tabloului indica spre un argument din linia de comanda. Toate argumentele liniei de comanda sunt siruri de caractere.

- ❑ **argv[0]** indica spre numele programului,
- ❑ **argv[1]** indica spre primul argument din linia de comanda dupa numele programului, etc...
- ❑ **argv[argc-1]** indica spre ultimul argument in linia de comanda



- ❑ **argc** si ***argv[]** sunt folosite pentru a prelua argumentele liniei de comanda
- ❑ Functia **main()** returneaza 0 daca executia programului decurge fara erori



TEST

1. In ce director se gasesc fisierele executabile create cu Codeblocks in C/C++ ?

Ex. Daca se creeaza un proiect : **nume**

Fisier cod: C:/test/pclp2/**nume**/main.c

Fisier executabil: C:/test/pclp2/**nume**/bin/Debug/**nume.exe**

2. Cum se executa un fisier din linia de comanda?

Raspuns

1. **/bin/Debug**

2. Se deschide aplicatia cmd, sau Command Prompt:

cd= change directory

cd.. Schimba directorul in directorul radacina

Se schimba directorul curent in directorul: **nume/bin/Debug**

In linia de comanda se scrie : **nume_fis_executabil** argumente de intrare

Nu este necesara extensia **.exe**, argumentele se separa cu spatiu

Cap 3: Linia de comanda

3.2. Argumentele functiei main()

EXEMPLE

Ex. 1: Programul `nume.exe` afiseaza Salut si numele din linia de comanda

```
//nume.exe
#include <stdio.h>
int main (int argc, char *argv[])
{printf("Salut %s !", argv[1]);
return 0;}
```

Rezultate :

```
> nume Radu
> Salut Radu !
```

1. Ce se intampla daca introducem un nume si prenume (3 sau mai multe argumente)?

R: Se afiseaza doar primul nume scris dupa numele fisierului executabil, restul se ignora

2. Cum afisam "Salut nume prenume" ?

R: `printf("Salut %s %s !", argv[1], argv[2]);`

3. Cum afisam "Salut nume1 nume2 prenume1 prenume2" ?

R: utilizam bucla for , while sau do while-> ex urmator

Cap 3: Linia de comanda

3.2. Argumentele functiei main()

EXEMPLE

Ex. 2: Programul `afiseaza.exe` afiseaza argumentele din linia de comanda

```
//afiseaza.exe
#include <stdio.h>
int main (int argc, char *argv[])
{int i;
printf("Salut ");
for (i=1;i<argc;i++)
    printf("%s ", argv[i]);
return 0;
}
```

Rezultate afisate:

```
>afiseaza Popa Mircea Radu
>Salut Popa Mircea Radu
```

Cap 3: Linia de comanda

3.2. Argumentele functiei main()

EXEMPLE

Ex.3. :Testare nr. de argumente din linia de comanda.Programul buna.exe afiseaza un text “Buna ziua <<numele>> ce mai faci ?” unde linia de comanda este de ex.:

> buna Ana

unde programul **buna.exe** are codul:

```
#include <stdio.h>
#include <stdlib.h>
int main(int argc, char *argv[])
{ if (argc!=2)
  { printf(“ati uitat sa scrieti numele!”);
    exit(1);  }
  printf(“Buna ziua  %s ce mai faci?”, argv[1]); return 0; }
```

Rezultate afisate:

> buna

> ati uitat sa scrieti numele!

Rezultate afisate:

> buna Ana

> Buna ziua Ana ce mai faci?

Cum afisam un nume si un prenume?

... if(argc!=3)..

```
printf(“Buna ziua  %s %s ce mai faci?”, argv[1], argv[2]);
```

Cap 3: Linia de comanda

3.2. Argumentele functiei main()

EXEMPLE

Ex. 4: Programul `suma.exe` aduna 2 nr. specificate in linia de comanda

```
//Programul suma.exe
#include <stdio.h>
#include <stdlib.h>
int main (int argc, char *argv[])
{int a,b;
if (argc!=3)
{ printf("Trebuie sa introduceti 2 nr.intregi \n"); exit(1); }
a=atoi(argv[1]); //convertirea sirului de caractere intr-un nr. intreg
printf("a=%d",a);
b=atoi(argv[2]);
printf(" b=%d",b);
printf(" s= %d",a+b); return 0;}
```

Rezultate afisate:

› Suma 12
› Trebuie sa introduceti 2 nr. intregi

Rezultate afisate:

› Suma 12 256
› a=12 b=256 s=268

Ce instructiuni adaugam ca sa afisam a^b ?

```
#include <math.h>
printf("pow= %d",pow(a,b));
```

Cum adunam 10 sau mai multe numere din linia de comanda?

-> utilizand un tablou si o bucla
for, while, sau do while

Cap 3: Linia de comanda

3.2. Argumentele functiei main()

EXEMPLE

Ex. 5: Programul `sum.exe` aduna un sir de nr.din linia de comanda

```
//Programul sum.exe
#include <stdio.h>
#include <stdlib.h>
int main (int argc, char *argv[])
{int i,s=0, n;
if (argc==1)      {printf("Nu ati introdus nici un nr!\n"); exit(1); }
else
{for (i=1;i<argc;i++) {    n=atoi(argv[i]);    s+=n;}
printf("suma argumentelor din linia de comanda este=%d",s);} return 0;}
```

Rezultate afisate:

```
> Sum 1 2 3 4 5 6
> suma argumentelor din linia de comanda: 21
```

Rezultate afisate:

```
> Sum
> Nu ati introdus nici un nr!
```

Cum calculam suma radicalului din fiecare numar introdus de la tastatura?

```
#include <math.h>    double n,s;
... n=sqrt(atof(argv[i])); s+=n // sau n=atof(argv[i]) ;s+=sqrt(n);
```

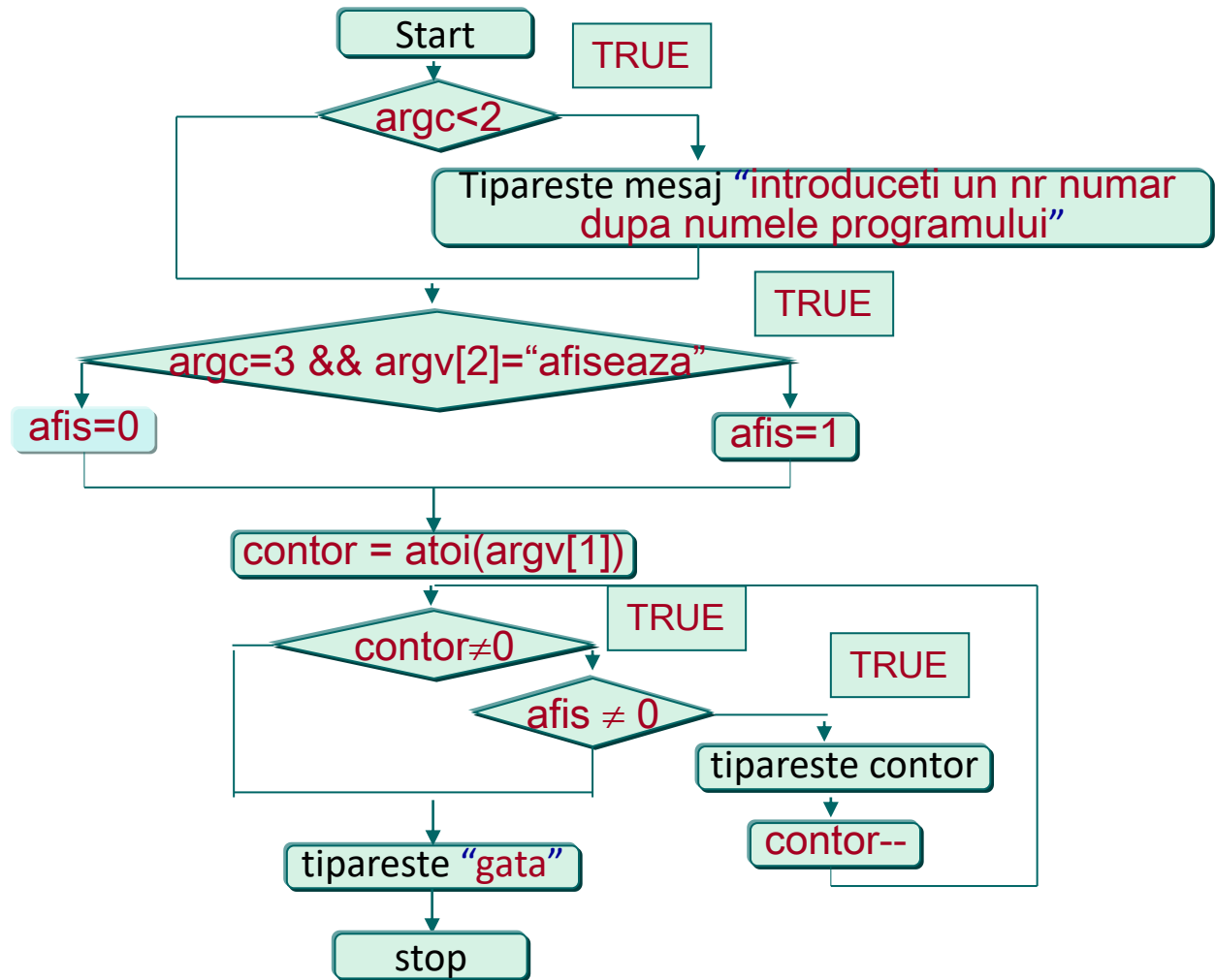
Cap 3: Linia de comanda

3.2. Argumentele functiei main()

EXEMPLE

Ex.6. : Programul `invers.exe` afiseaza numaratoarea inversa (pina la 0) a nr. specificat in linia de comanda, daca dupa nr. se introduce si textul (argumentul) "afiseaza"

Linie de comanda:
> `invers 5 afiseaza`



Cap 3: Linia de comanda

3.2. Argumentele functiei main()

EXEMPLE

Ex.6. : Programul `invers.exe` afiseaza numaratoarea inversa (pina la 0) a nr. specificat in linia de comanda, daca dupa nr. se introduce si textul (argumentul) "afiseaza":

```
//programul invers.exe
#include <stdio.h>
#include <stdlib.h>
#include <ctype.h>
#include <string.h>
int main (int argc, char *argv[])
{ int afis, contor;
  if (argc<2)
    { printf("introduceti un numar dupa numele programului\n"); exit(1);}
  if (argc==3 && !strcmp(argv[2], "afiseaza")) afis=1;
  else afis = 0;
  for (contor=atoi(argv[1]);contor;--contor)
    if(afis) printf("%d\n", contor);
    printf("gata\n"); return 0;}
```

Rezultate afisate:

```
> invers 5 afiseaza
> 5
> 4
> 3
> 2
> 1
> 0
> gata
```

Rezultate afisate:

```
> invers 5
> gata
```

Rezultate afisate:

```
> invers
> introduceti un numar dupa numele programului
```



TEST

Care va fi valoarea parametrului `argc` în funcția:
`int main (int argc, char *argv[])`,
dacă lansarea în execuție a programului se face cu linia de comandă de
mai jos?

`>program Ai trecut examenul?`

- a) 4
- b) 5
- c) 2
- d) "true"

Raspuns correct

a)



TEST

Care din variantele de mai jos este corectă atunci când lansarea în execuție a programului se face cu linia de comandă (1p):

>nume Program de test!

a) argc=4, argv[2]= "nume"

c) argc=4, argv[1]= "Program"

b) argc=5, argv[2]= "de"

d) argc=5, argv[1]= "nume"

Raspuns correct

c)



TEST

Care din variantele de mai jos este corecta atunci cand lansarea in executie a programului se face cu linia de comanda:

>program examen test PCLP2 !

a) argc=4, argv[2]= "examen"

c) argc=6, argv[1]= "examen"

b) argc=5, argv[2]= "test"

d) argc=4, argv[5]= "!"

Raspuns correct

b)



TEST

Ce se va afisa daca linia de comanda contine:

```
> Suma 10 20 30
```

```
//suma.exe
```

```
#include <stdio.h>
```

```
int main (int argc, char *argv[])
```

```
{ int j ;
```

```
j=atoi(argv[1])+atoi(argv[2]);
```

```
printf(“%d”,j);}
```

a) 60

b) 30

c) 10

d) 50

Raspuns correct

b)



TEST kahoot

Pentru login, introduceti codul afisat pe ecran, in browser la adresa:

<http://kahoot.it>