

## Laborator 5

### Linia de comandă. Argumentele funcției main()

În acest capitol sunt prezentate considerații teoretice privind utilizarea liniei de comandă și argumentele funcției main() .

#### CONSIDERAȚII TEORETICE

Programele executabile rezultate în urma compilării și link-editării fișierelor text într-un mediu de programare sunt fișiere care se pot executa direct din linia de comandă într-o fereastră de tip Command Prompt.

**Linia de comandă** este compusă din numele programului ce urmează a fi executat și argumentele liniei de comandă. Argumentele liniei de comandă sunt informațiile ce urmează numelui de program care va fi executat. Argumentele sunt despărțite prin câte un singur spațiu și reprezintă argumentele funcției main din programul executabil specificat în linia de comandă.

Headerul funcției main() este :

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

unde:

- argc** = este de tip întreg și memorează numărul de argumente din linia de comandă. Are valoarea 1 implicit (cel puțin numele programului este dat în linia de comandă).
- argv** = este un tablou (șir) de pointeri către șiruri de caractere. Fiecare element al tabloului indică spre un argument din linia de comandă. Toate argumentele liniei de comandă sunt șiruri de caractere.

Astfel:

- **argv[0]** indică spre numele programului,
- **argv[1]** indică spre primul argument din linia de comandă după numele programului, etc...
- **argv[ ]** parantezele drepte fără conținut indică faptul că șirul are lungimea nedeterminată.

Argumentele funcției main() , argc și argv[] sunt folosite pentru a prelua argumentele liniei de comandă.

#### PROBLEME REZOLVATE

*Ex.1 Programul preia din linia de comandă un nume specificat de utilizator și afișează mesajul "Hello <<nume>> ce mai faci? " Proiectul se va numi nume.c și va deveni în urma compilării și link-editării "nume.exe". Acest program executabil se va lansa de la prompterul sistem ca o linie de comandă, astfel :*

```
>nume Mihai
```

Varianta in C	Varianta in C++
<pre>#include &lt;stdio.h&gt; #include &lt;stdlib.h&gt; int main (int argc, char *argv[]) {if (argc!=2)     {printf("Ati uitat sa va scrieti numele: \n");     exit(1);} printf("Hello %s ce mai faci?", argv[1]); return 0;}</pre>	<pre>#include &lt;iostream&gt; #include&lt;stdlib.h&gt; using namespace std; int main (int argc, char *argv[]) {if (argc!=2)     {cout&lt;&lt;"Ati uitat sa va scrieti numele: "&lt;&lt;endl;     exit(1);} cout&lt;&lt;"Hello "&lt;&lt;argv[1]&lt;&lt;" ce mai faci?"; return 0;}</pre>

#### Rezultate:

```
D:\laura\codeblocks\nume\bin\Debug>nume Laura
Hello Laura ce mai faci?
```

```
D:\laura\codeblocks\nume\bin\Debug>nume
Ati uitat sa va scrieti numele:
```

#### Aplicație:

Să se modifice programul astfel încât să se introducă din linia de comandă: numele, prenumele și vârsta.

**Ex.2. Programul citește un număr din linia de comandă și realizează numărătoarea inversă pornind de la numărul specificat până la 0 (similar cu cronometru). La sfârșit se afișează textul "gata". Exemplu de linie de comandă: >afiseaza 5 invers**

Varianta in C	Varianta in C++
<pre>#include &lt;stdio.h&gt; #include &lt;stdlib.h&gt; #include &lt;ctype.h&gt; #include &lt;string.h&gt; int main (int argc, char *argv[]) {int afis, contor; if (argc&lt;2) {printf("introduceti un numar intreg, mai incercati o data\n"); exit(1);} if (argc==3 &amp;&amp; !strcmp(argv[2], "invers")) afis=1; else afis = 0; for (contor=atoi(argv[1]);contor;--contor) if(afis) printf("%d\n", contor); printf("gata\n"); return 0;}</pre>	<pre>#include &lt;iostream&gt; #include &lt;stdlib.h&gt; #include &lt;ctype.h&gt; #include &lt;string.h&gt; using namespace std; int main (int argc, char *argv[]) {int afis, contor; if (argc&lt;2) {cout&lt;&lt;"introduceti un numar intreg, mai incercati o data"&lt;&lt;endl; exit(1);} if (argc==3 &amp;&amp; !strcmp(argv[2], "invers")) afis=1; else afis = 0; for (contor=atoi(argv[1]);contor;--contor) if(afis) cout&lt;&lt;contor&lt;&lt;endl; cout&lt;&lt;"gata"&lt;&lt;endl; return 0;}</pre>

#### Rezultate:

```
D:\laura\codeblocks\afiseaza\bin\Debug>afiseaza
introduceti un numar intreg, mai incercati o data
```

#### Aplicație:

Să se modifice programul astfel încât să se introducă din linia de comandă: un număr întreg și să se afișeze dacă numărul este în intervalul [5,100] sau nu.

```
D:\laura\codeblocks\afiseaza\bin\Debug>afiseaza 5 inv
5
4
3
2
1
gata
```

**Ex.3. Programul suma.c calculează și afișează suma a 2 numere preluate din linia de comandă.**

**Exemplu de linie de comandă:**

**>suma 200 350**

<pre>#include &lt;stdio.h&gt; #include &lt;stdlib.h&gt; int main (int argc, char *argv[]) {int a,b; if (argc!=3) { printf("Trebuie sa introduceti 2 nr.intregi \n"); exit(1); } //convertirea sirului de caractere la nr. intreg a=atoi(argv[1]); printf("a=%d ",a); b=atoi(argv[2]); printf("b=%d ",b); printf("s= %d ",a+b); return 0;}</pre>	<pre>#include &lt;iostream&gt; #include &lt;stdlib.h&gt; using namespace std; int main (int argc, char *argv[]) {int a,b; if (argc!=3) { cout&lt;&lt;"Trebuie sa introduceti 2 nr.intregi "&lt;&lt;endl; exit(1); } //convertirea sirului de caractere la nr. intreg a=atoi(argv[1]); cout&lt;&lt;"a="&lt;&lt;a; b=atoi(argv[2]); cout&lt;&lt;" b="&lt;&lt;b; cout&lt;&lt;" s="&lt;&lt;a+b; return 0;}</pre>
---	--

**Rezultate:**

```
D:\laura\codeblocks\suma\bin\Debug>suma
Trebuie sa introduceti 2 nr.intregi
```

```
D:\laura\codeblocks\suma\bin\Debug>suma 100 20
a=100 b=20 s= 120
```

**Aplicație:**

Să se modifice programul astfel încât să se efectueze și alte operații aritmetice (scăderea, înmulțirea, împărțirea, etc) introduse în linia de comandă.

**Ex.4. Programul calculează și afișează suma unui șir de numere preluate din linia de comandă.**

**Exemplu de linie de comandă:**

**>sum 1 2 4 6 8**

<pre>#include &lt;stdio.h&gt; #include &lt;stdlib.h&gt; 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&lt;argc;i++) {n=atoi(argv[i]);s+=n; } printf("suma argumentelor din linia de comanda este=%d",s); }return 0; }</pre>	<pre>#include &lt;iostream&gt; #include &lt;stdlib.h&gt; using namespace std; int main (int argc, char *argv[]) {int i,s=0, n; if (argc==1) {cout&lt;&lt;"Nu ati introdus nici un nr!"&lt;&lt;endl; exit(1);} else {for (i=1;i&lt;argc;i++) {n=atoi(argv[i]);s+=n; } cout&lt;&lt;"suma argumentelor din linia de comanda este="&lt;&lt;s;} return 0;}</pre>
--	---

### Rezultate:

```
D:\laura\codeblocks\sum\bin\Debug>sum
Nu ati introdus nici un nr!
```

```
suma argumentelor din linia de comanda este=150
D:\laura\codeblocks\sum\bin\Debug>
```

### Aplicație:

Să se modifice programul astfel încât să se afișeze produsul numerelor și suma pătratelor numerelor introduse în linia de comandă.

## PROBLEME PROPUSE

1. Să se scrie un program care să ordoneze un șir de nr. întregi introdus în linia de comandă.
2. Să se scrie un program care inversează șirul de caractere introdus în linia de comandă.
3. Să se scrie un program care concatenează două șiruri de caractere introduse în linia de comandă
4. Să se scrie un program care determină minimul și maximul precum și elementele impare dintre elementele unui șir de numere întregi specificate în linia de comandă.
5. Să se scrie programul care determină și afișează valoarea funcției  $\ln(1+x)$  într-un punct dat  $x \in (-1,1]$  introdus în linia de comandă, utilizând dezvoltarea în serie :

$$\ln(1+x) = \frac{x}{1} - \frac{x^2}{2} + \frac{x^3}{3} - \frac{x^4}{4} + \dots$$

și să se compare rezultatele obținute astfel cu rezultatul obținut prin apelarea funcției  $\log()$  din biblioteca `<math.h>`.

6. Sa se scrie un program care citește din linia de comanda un șir  $x[i]$ ,  $i=1, n$  ( $n$  întreg) numere reale și calculează și afișează valorile funcției :  $f(x) = \begin{cases} x^2 - 1, & x > 0 \\ -5, & x < 0 \end{cases}$