

Șiruri de caractere

Bază

- **Declararea** unui șir de caractere de maxim 200 de caractere: `char sir[201];`
 - Șirul declarat mai sus poate stoca 200 de caractere, plus delimitatorul de sfârșit de șir de caractere (`'\0'`).
- **Declararea unui pointer** la șir de caractere: `char *p;`
- **Citirea** unui șir de caractere de la tastatură (pentru fișiere înlocuiește `cin` cu streamul de intrare):
 - Citirea unui **cuvânt, până la primul spațiu**: `cin>>sir;`
 - Citire de **linie întreagă, cel mult 200 de caractere, ce poate avea spații, până la linie nouă**:

```
cin.get(sir, 201); // citirea propriu-zisă
cin.get(); // trece de următorul caracter, de regulă '\n'
```

 - * Alternativ, aceste 2 funcții pot fi combinate într-una: `cin.getline(sir, 201);`
- **Afișarea** unui șir de caractere pe ecran (pentru fișiere înlocuiește `cout` cu streamul de ieșire):
`cout<<sir;`

Funcții utile

`#include <cstring>` sau `#include <string.h>` pentru a accesa funcțiile de mai jos ce prelucrează șiruri de caractere. Copierea și concatenarea dintr-un șir în același șir merge doar pentru bacalaureat.

- `strlen(sir)`: lungimea șirului.
- `strcpy(dest, sursa)`: înlocuiește conținutul din `dest` cu cel din `sursa`.
 - Pentru ștergerea dintr-un șir de caractere a `x` caractere începând cu poziția `p` se poate folosi `strcpy(sir+p, sir+p+x);`
- `strncpy(dest, sursa, n)`: copiază în `dest` primele `n` caractere din `sursa` pe primele `n` poziții.
- `strcmp(sir1, sir2)`: compară cele două șiruri de caractere și întoarce:
 - 0 dacă șirurile sunt identice.
 - <0 dacă `sir1<sir2`.
 - >0 dacă `sir1>sir2`.
- `strncmp(sir1, sir2, n)`: la fel ca `strcmp`, dar limitat la primele `n` caractere ale șirurilor.
- `stricmp(sir1, sir2)`: la fel ca `strcmp`, dar nu face diferențierea între litere mici și mari.
- `strcat(dest, sursa)`: alipește la capătul lui `dest` conținutul din `sursa`.
- `strncat(dest, sursa, n)`: alipește la capătul lui `dest` primele `n` caractere din `sursa`.
- `strchr(sir, c)`: pointer către prima apariție a caracterului `c` în `sir` (NULL dacă nu e cazul).
 - Pentru a verifica dacă caracterul `c` este vocală se poate folosi `strchr("aeiou", c)!=NULL`.

- `strstr(sir1, sir2)`: pointer către prima apariție a lui `sir2` în `sir1` (NULL dacă nu e cazul).
- `strtok(sir, sep)`: pointer cu primul subșir din `sir` delimitat de caractere din `sep`. `sir` devine acest prim subșir.
- `strtok(NULL, sep)`: pointer cu următorul subșir din `sir` delimitat de caractere din `sep` (NULL dacă nu e cazul).