

# Fundamentele programării

## Curs 8

**Şef lucr.dr.ing. GENGE Béla**

Universitatea “Petru Maior”, Departamentul de Informatică  
Tîrgu Mureş, România  
bela.genge@ing.upm.ro

- Secvențe de caractere terminate cu '\0' (caracterul nul, terminatorul de șir)
- Sintaxa de declarare:

```
char nume[dim];
```

- `char str[] = "sir caractere";`
- `char str[50] = "sir caractere";`
- `char* str = "sir caractere";`
- `const char* str = "sir caractere";`
- `char* const str = "sir caractere";`
- `const char* const str = "sir caractere";`

- scanf/printf: format '%s'
- gets/puts: nu necesită format (string.h)
- cin/cout: determinarea automată a tipului

### Exemplu

```
char str[200];  
scanf("%s", str); //Citire până la primul spațiu  
puts(str);
```

- `int strlen(const char* str)`
- `char* strcpy(char* strDest, const char* strSursa)`
- `int strcmp(const char* str1, const char* str2)`

# Probleme

- 1. Să se determine lungimea unui șir de caractere (fără utilizarea funcției `strlen`) utilizând operații cu pointeri.
- 2. Să se afișeze caracterele unui șir citit de la tastatură pe linii separate.
- 3. Să se adauge caracterul '+' la sfârșitul unui șir de caractere.
- 4. Să se concateneze două șiruri de caractere citite de la tastatură.
- 5. Să se determine dacă un șir este conținut într-un alt șir de caractere.
- 6. Să se implementeze un program ce afișează toate cuvintele dintr-o linie citită de la tastatură. Cuvintele pot fi delimitate de următoarele caractere: ' ', ',', ';', ':', '.', ..

## Șiruri de caractere în C++

- Tipul(clasa) `string`
- Declaraire: `string str;`
- Inițializare: `string str = "Text";`
- Accesare buffer: `str.c_str()`
- Afișare: `cout << str.c_str();`
- Concatenare: `str+="Text";`
- Accesare caractere: `str[0]='N';`
- Lungimea: `str.length();`

- Să se citească un string de la tastatură
- Să se modifice minusculele în majuscule și majusculele în minuscule
- Să se declare un vector de 100 de string-uri
- Să se citească de la tastatură N string-uri în vectorul declarat
- Să se afișeze cel mai scurt string



## Funcții de lucru cu memoria

- `void * memset(void * ptr, int value, size_t num)`
- `void * memcpy(void * destination, const void * source, size_t num)`
- Fie un șir de caractere. Să se șteargă ultimele 5 caractere folosind funcția `memset`.
- Să se copieze un tablou de întregi într-un alt tablou de întregi folosind funcția `memcpy`.