



MINISTERUL EDUCAȚIEI ȘI CERCETĂRII
ROMÂNIA

UNIVERSITATEA DE MEDICINĂ,
FARMACIE, ȘTIINȚE ȘI TEHNOLOGIE
„GEORGE EMIL PALADE”
DIN TÂRGU MUREȘ

FACULTATEA DE INGINERIE ȘI TEHNOLOGIA INFORMAȚIEI

Algoritmi fundamentali

Curs 9

Operații cu șiruri de caractere

Dr. ing. Kiss Istvan

istvan.kiss@umfst.ro

Cuprins

1. Referire la un element
2. Tablou (vector) de siruri de caractere
3. Subalgoritmi de prelucrare
 1. lungime, copiere, concatenare, comparare, inserare, cautare, stergere, tokenizare
4. Subalgoritmi de conversie
 1. din sir in numar si invers
5. Probleme

"string"

's'	't'	'r'	'i'	'n'	'g'	'\0'
-----	-----	-----	-----	-----	-----	------

'c'

'c'

"c"

'c'	'\0'
-----	------

1. Intro

- Declarare, initializare in pseudocod:

string sir;

string sir="Helo world!";

character sir[50];

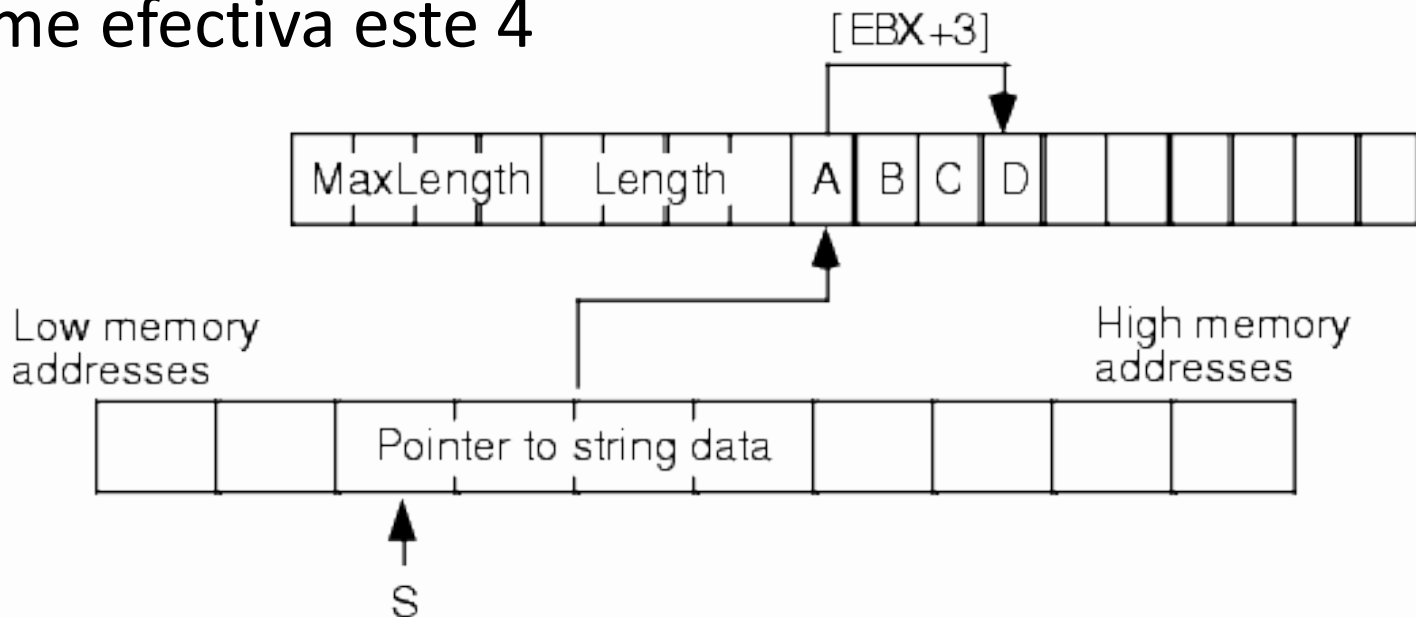
citeste sir; → citire sir de caractere de pe tastatura.

afiseaza sir; → scrie pe ecran continutul sirului.

- In practica, in memoria calculatorului sirurile de caractere se termina cu caracterul '\0' (NULL).

1. Referire la un element

- Elementul intr-un sir de caractere este un singur caracter.
- string s:="ABCD"
- s[0]='A'
- s[4]='\0'
- lungime efectiva este 4



2. Tablou de siruri de caractere

- `string tabel[15];`
- Este o matrice de caractere!

pentru $i:=0,14$ exec.

afiseaza `tabel[i];`

`sf.pentru`

3. Subalgoritmi de prelucrare

- **Lungimea unui sir**

```
string sir:="Caractere";
```

```
intreg lungime:=length(sir);
```

```
functie length(string s)
```

```
    len:=0;
```

```
    cattimp s[len]<>'\'0' exec.
```

```
        len:=len+1;
```

```
    sf.cat.
```

```
    returneaza len;
```

```
sf.length
```

3. Copierea unui sir

subprogramul copi(string dest, string sursa)

i:=0;

cattimp sursa[i]<>'\\0' exec.

dest[i]:=sursa[i];

i:=i+1;

sf.cat.

sf.copi.

Apel: copi(sir1, "informatica") → sir1="informatica"

Se poate si cu structura pentru, prin apelarea functiei length!

Alta versiune: functia copin(dest,sursa,n) – copiaza numai n caractere din sursa!!! si returneaza restul caracterelor din sursa

3. Concatenarea unui sir

subprogramul cat(string dest, string sursa)

i:=0; lungDest := length(dest);

cattimp sursa[i]<>'\'0' exec.

dest[lungDest+i]:=sursa[i];

i:=i+1;

sf.cat.

sf.cat.

Apel: cat("Helo", " World") → "Hello World"

3. Compararea sirurilor

functie compar(string sir1, string sir2)

pos1:= 0; pos2:=0;

lungSir1 := length(sir1);

lungSir2 := length(sir2);

cattimp (pos1<lungSir1) sau (pos2<lungSir2) exec.

daca pos1==lungSir1 atunci returneaza -1; sf.daca

daca pos2==lungSir2 atunci returneaza 1; sf.daca

daca sir1[pos1] < sir2[pos2] atunci returneaza -1; sf.daca

daca sir1[pos1] > sir2[pos2] atunci returneaza 1; sf.daca

pos1++; pos2++;

sf.cat.

returneaza 0;

sf.compar.

Functia compar nu este un algoritm pentru masurarea similitudinii cuvintelor!!!

Apel: compar("abc", "bcd") → -1

3. Inserare character sau sir

- Prin mutarea caracterelor la dreapta.

```
string sir:="In este pauza";
```

```
string sursa:="curand ";
```

```
string aux, aux1;
```

```
aux1:= copin(aux,sir,3);
```

```
cat(aux,sursa);
```

```
cat(aux, aux1); → aux va fi "In curand este pauza".
```

Tema: sa se scrie o functie: `inserare(sir, sursa, poz);`

3. Cautare intr-un sir

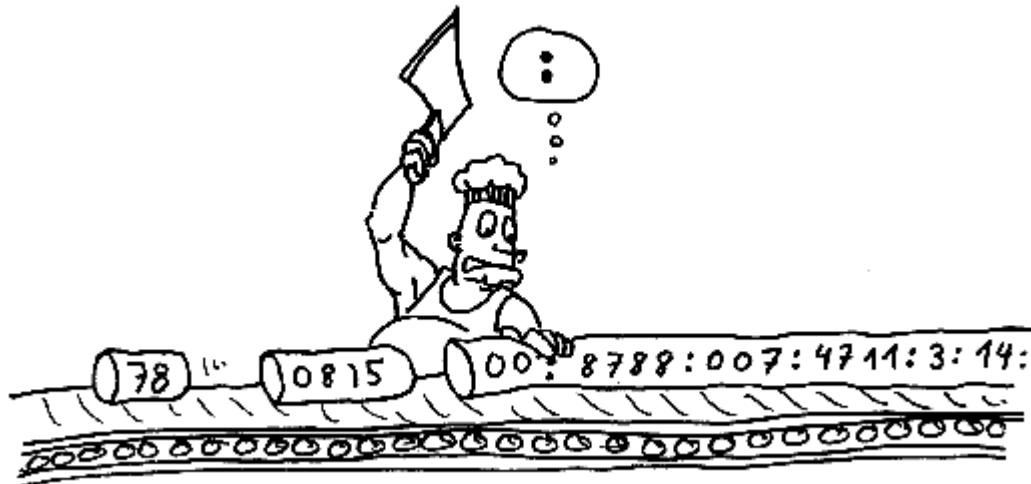
- functie `cauta(string sir, string cheie)...`
 - daca cheie exista, returneaza pozitia de inceput a sirului cheie
 - in caz contrar, returneaza -1

3. Stergere dintr-un sir

- Ramane pentru problema de rezolvat...!

3. Tokenizare

- Reprezinta spargerea unui sir in atomi lexicali (in cuvinte)
- Este util in extragerea de cuvinte dintr-un sir in care cuvintele sunt separate prin ‘,’ sau ‘;’ de ex.
- Solutie: (prin parcurgerea sirului)
 1. Se copiaza caracterele in primul element al unui vector de *string*, pana cand nu se intalneste token-ul.
 2. Se copiaza caracterele de dupa token in al doilea element al vectorului, pana cand nu se intalneste token-ul.
 3. Se repeta pana la terminarea caracterelor din sir.



4. Subalgoritmi de conversie

- **Conversie din sir in numar, ex.alg.:**

a:=0;

string sir:="15238"

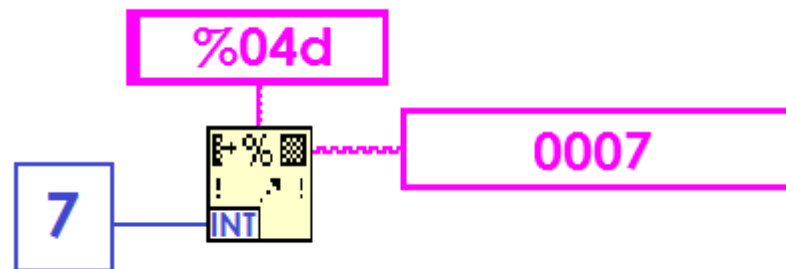
pentru caracter din sir exec.

a:=10*a+getCifra(character); //caracter trebuie convertit
in cifra!!!

sf.pentru

- **Conversie numar in sir de caractere:**

- ex. alg.: exista 10 cif
cifra...



5. Probleme

1. Se citește un șir de caractere format din cel mult 100 de caractere. Să se numere și să se șteargă vocalele din șir. Ex: Pentru șirul abracadabra se afișează 5 si brcdbr.
2. Se citește un cuvânt a cu cel mult 100 de litere. Să se afișeze pe linii separate toate cuvintele care se pot forma prin eliminarea unei singure litere din cuvântul citit. Ex: dacă se citește cuvântul alin se vor afișa: lin, ain, aln, ali.

5. Probleme

3. Se citește un cuvânt format din număr par de litere. Să se interschimbe litera de pe prima poziție cu cea de pe a doua, a treia cu a patra, etc. Ex. cosmin => ocmsni.
4. Se citesc 2 cuvinte a și b. Să se determine dacă sunt anagrame.
5. Să se determine numărul aparițiilor unui subsir într-un sir.