

PROGRAMARE PROCEDURALĂ – LABORATOR NR. 8–

1. Scrieți un program care citește n cuvinte și afișează toate cuvintele de lungime maximă citite. De exemplu, dacă se vor citi cuvintele "mere", "tei" și "pere" se vor afișa cuvintele "mere pere".

Varianta 1:

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

int main()
{
    char cuvinte[100][21];
    int n;

    printf("n = ");
    scanf("%d", &n);

    int lMax = 0;
    printf("Cuvintele:\n");
    for(int i = 0; i < n; i++)
    {
        scanf("%s", cuvinte[i]);

        if(strlen(cuvinte[i]) > lMax)
            lMax = strlen(cuvinte[i]);
    }

    printf("Cuvintele cu lungimea maxima:\n");
    for(int i = 0; i < n; i++)
        if(strlen(cuvinte[i]) == lMax)
            printf("%s ", cuvinte[i]);

    return 0;
}
```

Varianta 2:

Fie $n = 6$, cuvintele "mere", "Ana", "maci", "prune", "are", "multe"

Cuvântul curent	Lungimea maximă curentă a unui cuvânt	Șirul format din cuvintele de lungimea maximă curentă
—	0	""
"mere"	4	"mere"
"Ana"	4	"mere"
"maci"	4	"mere maci"
"prune"	5	"prune"
"are"	5	"prune"
"multe"	5	"prune multe"

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

int main()
{
    //presupunem ca un cuvânt are maxim 20 de caractere
    char cuvânt[21];
    //presupunem ca se vor citi maxim 100 de cuvinte
    char cuvinte_lmax[2100];
    int n;

    printf("n = ");
    scanf("%d", &n);

    int lMax = 0;
    strcpy(cuvinte_lmax, "");

    printf("Cuvintele:\n");
    for(int i = 0; i < n; i++)
    {
        scanf("%s", cuvânt);

        if(strlen(cuvânt) == lMax)
            strcat(strcat(cuvinte_lmax, " "), cuvânt);
        else
            if(strlen(cuvânt) > lMax)
            {
                lMax = strlen(cuvânt);
                strcpy(cuvinte_lmax, cuvânt);
            }
    }

    printf("Cuvintele cu lungimea maxima: %s\n", cuvinte_lmax);

    return 0;
}
```

2. Scrieți un program care afișează toate cuvintele dintr-o propoziție în ordinea descrescătoare a lungimilor. Dacă mai multe cuvinte au aceeași lungime, ele se vor afișa în ordine alfabetică. De exemplu, pentru propoziția "Ana are pere și mere multe!" se vor afișa cuvintele în ordinea "multe", "mere", "pere", "Ana", "are", "și".

Rezolvare:

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

int main () {
    char prop[501];
    char cuvinte[50][21];
    char* ptr;
    int contorCuv = 0;

    printf("Introduceti o propzitie (max 500 char):\n");
    fgets(prop, sizeof(prop), stdin);

    ptr = strtok(prop, " ,.?!?\n");
    while (ptr != NULL) {
        strcpy(cuvinte[contorCuv], ptr);
        contorCuv++;
        ptr = strtok(NULL, " ,.?!?\n");
    }

    for (int i = 0; i < contorCuv; i++)
        for (int j = i + 1; j < contorCuv; j++)
            if (strlen(cuvinte[i]) < strlen(cuvinte[j]))
            {
                char aux[21];
                strcpy(aux, cuvinte[j]);
                strcpy(cuvinte[j], cuvinte[i]);
                strcpy(cuvinte[i], aux);
            }
        else
            if (strlen(cuvinte[i]) == strlen(cuvinte[j]))
                if (strcmp(cuvinte[i], cuvinte[j]) > 0)
                {
                    char aux[21];
                    strcpy(aux, cuvinte[j]);
                    strcpy(cuvinte[j], cuvinte[i]);
                    strcpy(cuvinte[i], aux);
                }

    for (int i = 0; i < contorCuv; i++)
        printf("%s\n", cuvinte[i]);
    return 0;
}
```

3. Scrieți o funcție care primește un șir de caractere de forma "x+y", unde x și y sunt două numere naturale, și furnizează șirul "x+y=s", unde s este suma celor două numere x și y.

Rezolvare:

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

char* suma(char* sir)
{
    char aux[51];
    int x, y;

    sscanf(sir, "%d+%d", &x, &y);

    int suma = x + y;

    sprintf(aux, "%d+%d=%d", x, y, suma);

    char* ptr = malloc(strlen(aux) + 1);
    strcpy(ptr, aux);

    return ptr;
}

int main ()
{
    char sir[51];
    char* ptrSir;

    printf("Introduceti un sir de forma x+y: \n");
    scanf("%s", sir);

    ptrSir = suma(sir);

    printf("Rezultatul este: %s\n", ptrSir);

    free(ptrSir);

    return 0;
}
```