

Curs 10: Program ilustrativ inițial

```
#include<iostream>
using namespace std;
const int dim = 100;
void dubleazaLit(char text[dim], char lit){
    //dubleaza orice aparitie a caracterului lit
    char aux[dim];
    int j = 0;
    for (int i = 0; i < dim && (aux[j] = text[i]) != '\0'; i++, j++){
        if (aux[j] == lit) aux[++j] = lit;
    }
    for (int i = 0; i < dim && (text[i] = aux[i]) != '\0'; i++){}
}
void dubleazaLitPeLoc(char text[dim], char lit){
    for (int i = 0; i < dim && text[i] != '\0'; i++){
        if (text[i] == lit){
            // inseram lit cu scoatere pe loc
            i++;
            char chStg = lit;
            for (int j = i; j < dim; j++){
                char chDrp = text[j]; //scoatem cu dreapta
                text[j] = chStg; //punem cu stanga
                if (chStg == '\0') break;
                chStg = chDrp; //trecem din dreapta in stanga
            }
        }
    }
}
void dubleazaLitPeLoc2(char text[dim], char lit){
    for (int i = 0; i < dim && text[i] != '\0'; i++){
        if (text[i] == lit){
            // inseram lit cu scoatere in avans
            i++;
            char chStg = text[i], chDrp=text[i+1];
            text[i] = lit;
            for (int j = i +1; j < dim && (text[j] = chStg) != '\0'; j++){
                chStg = chDrp;
                chDrp = text[j + 1];
            }
        }
    }
}
int main(){
    char text[dim] = "__a_aa_a_";
    cout << text << endl;
    dubleazaLitPeLoc(text, 'a');
    cout << text << endl;
    return 0;
}
/*REZULTAT:
__a_aa_a_
__aa_aaaa_aa_
Press any key to continue . . .*
```

Program ilustrativ final

```
#include<iostream>
using namespace std;

void afisareFaraDubluri(char text[]){
    cout << text[0];
    char ch;
    for (int i = 1; (ch = text[i]) != '\0'; i++){
        if (ch == text[i - 1]) continue;
        cout << ch;
    }
    cout << endl;
}

void afisareFaraDubluriSiStelute(char text[]){
    //cautam primul ch diferit de *
    int i = 0;
    for (; text[i] == '*'; i++){
    }
    if (text[i] == '\0') return;
    char chAfisat, ch;
    cout << (chAfisat = text[i]);
    for (i++; (ch = text[i]) != '\0'; i++){
        if (ch == chAfisat || ch == '*') continue;
        cout << (chAfisat = ch);
    }
    cout << endl;
}

void curataDubluri(char text[]){
    char ch;
    for (int i = 1; (ch = text[i - 1]) != '\0'; i++){
        //stabilim cate caractere trebuie eliminate
        int delta = 0;
        for (; ch == text[i + delta]; delta++){
        }
        if (delta == 0) continue;
        //eliminam delta caractere
        for (int j = i; (text[j] = text[j + delta]) != '\0'; j++){
        }
    }
}

void afisareDublata(char text[]){
    //stringul "abc" este afisat "aabbcc"
    char ch;
    for (int i = 0; (ch = text[i]) != '\0'; i++){
        cout << ch << ch;
    }
    cout << endl;
}
```

```

void dubleaza(char text[]) {
    //stringul "abc" este atransformat in "aabbcc"
    char chStg, chDrp;
    for (int i = 0; (chStg = text[i]) != '\0'; i += 2) {
        //facem loc dublurii lui text[i]
        int j = i + 1;
        for (; chStg != '\0'; j++) {
            chDrp = text[j];
            text[j] = chStg;
            chStg = chDrp;
        }
        text[j] = chStg;
    }
}

```

```

void dubleaza2(char text[]) {
    char chStg, chDrp;
    int i = 0;
    while (chDrp = text[i++]) {
        int j = i++;
        do {
            chStg = chDrp;
            chDrp = text[j];
        } while (text[j++] = chStg);
    }
}

```

```

int main(){
    char text[1000] = "ab*b";
    cout << text << endl;
    afisareDublata(text);
    dubleaza(text);
    cout << text << endl;
    dubleaza2(text);
    cout << text << endl;
    afisareFaraDubluri(text);
    afisareFaraDubluriSiStelute(text);
    curataDubluri(text);
    cout << text << endl;
    return 0;
}

```

```

/* MONITOR:
ab*b
aabb**bb
aabb**bb
aaaabbbb****bbbb
ab*b
ab
ab*b
Press any key to continue . . .*/

```