

```

/*****
SIMULAREA FUNCTIONARII UNUI AUTOMAT FINIT DETERMINIST

INPUT: cuvantul de analizat (sir de caractere terminat cu #)
OUTPUT: mesaj de recunoastere a cuvantului (Yes / No)

Descrierea automatului simulat
Alfabetul de intrare: {A,B,...,Z,a,...,z,0,1,...,9}
Multimea de stari:    {s0,s1}
Starea initiala:      {s0}
Stari finale:         {s1}
Functia de evolutie:  f(s0,l)=s1 {l litera}
                      f(s1,l)=f(s1,c)=s1 {l litera, c cifra}

Limbaj recunoscut:    identificatori C
                      {sir de litere si cifre ce incepe cu litera}
*****/
#include<stdio.h>
#include<CTYPE.H>

#define ERROR 200 /* codificare pentru caracter eronat la intrare */

char c; /* litera citita */
int stare; /* variabila ce contine starea curenta */
int coloana; /* coloana corespunzatoare caracterului citit
              litera = 0; cifra =1; */

int tabel[2][2]={ {1,ERROR} , {1,1} }; /* diagrama de stari */
int main()
{
    printf("\nintrodu cuvantul de analizat (terminat cu #):");

    stare = 0; /* initializare stare */

    while( ( c=getchar() ) != '#' )
    {
        /* analizez caracterul citit */
        if (isalpha(c)) coloana = 0;
        else if(isdigit(c)) coloana = 1;
        else { /* caracter ilegal in text */
            printf("caracter ilegal in text %c \n", c);
            stare = ERROR;
            break;
        };

        stare=tabel[stare][coloana]; /* calcul stare noua */

        if (stare==ERROR) break;
    }; /* end while*/

    /* analizez starea reziduala */
    if(stare==1) printf("cuvantul ESTE identificator\n");
}

```

```
    else printf("cuvantul NU ESTE identificator \n");  
}
```