

```

/*****
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 S0 0      /* codificarea starii s0 prin intregul 0 */
#define S1 1
#define ERROR 2   /* codificare pentru caracter eronat la intrare */

char c; /* litera citita */
int stare; /* variabila ce contine starea curenta */

int main()
{
    printf("\nintrodu cuvantul de analizat (terminat cu #):");

    stare = S0; /* initializare stare */

    while( ( c=getchar() ) != '#' )
    {
        /* analizez caracterul citit */
        switch (stare) {
            case S0: if( isalpha(c) ) stare=S1;
                     else {
                         stare=ERROR;
                         printf("EROARE: caracter ilegal in cuvant %c \n", c);
                     };
                     break;
            case S1: if( !(isalpha(c) || isdigit(c)) ) {
                         stare=ERROR;
                         printf("EROARE: caracter ilegal in cuvant %c \n", c);
                     };
                     break;
            default: break;
        }; /* end switch */
        if (stare==ERROR) break;
    }; /* end while*/

    /* analizez starea reziduala */

```

```
if(stare==S1) printf("cuvantul ESTE identificator\n");  
    else printf("cuvantul NU ESTE identificator \n");  
}
```