UNIVERSIDAD AUTÓNOMA DE CHIHUAHUA
Facultad de Ingeniería

Ingeniería en Ciencias de la Computación

TEORÍA DE LA COMPUTACIÓN
Programa AFD

Trabajo de: ADRIAN A. GONZÁLEZ DOMÍNGUEZ *[359834]*
Asesor: MARIO ANDRES CUEVAS GUTIERREZ

16 de septiembre de 2024

Teoría de la Computación | Lunes 16/Septiembre/2024

Programa AFD

Adrian González Domínguez

```
#include <stdio.h>
int automaton(void);
int choose_column(int alphabet_length, char alphabet[], char c);
int is_final(int finals_length, int finals[], int state);
char result[100];
int result_index;
FILE* input_file;
int main(){
  input_file=stdin;
  while(!feof(input_file)){
    result_index=0;
    if(automaton()){
      result[result_index='\0';
      printf("Cadena válida: %s\n", result);
    }else{
      printf("Cadena inválida\n");
    }
  }
  return 0;
}
```

```
// Automata A
int automaton(void){
 int states[][3]={
  {0, 1, 3}, //0
  {0, 2, 3}, //1
  {2, 2, 3}, //2
  {3, 3, 3}, //3
 };
 int finals[]={2};
 int finals_length=1;
 char alphabet[]={'a', 'b'};
 int alphabet_length=2;
 int state=0;
 char c;
 while(
  !feof(input_file) &&
  (c=getc(input_file))!='\n'
 ){
  result[result_index++]=c;
  int column=choose_column(
     alphabet_length, alphabet, c
  );
  state=states[state][column];
 }
 return is_final(
    finals_length, finals, state
 );
}
```

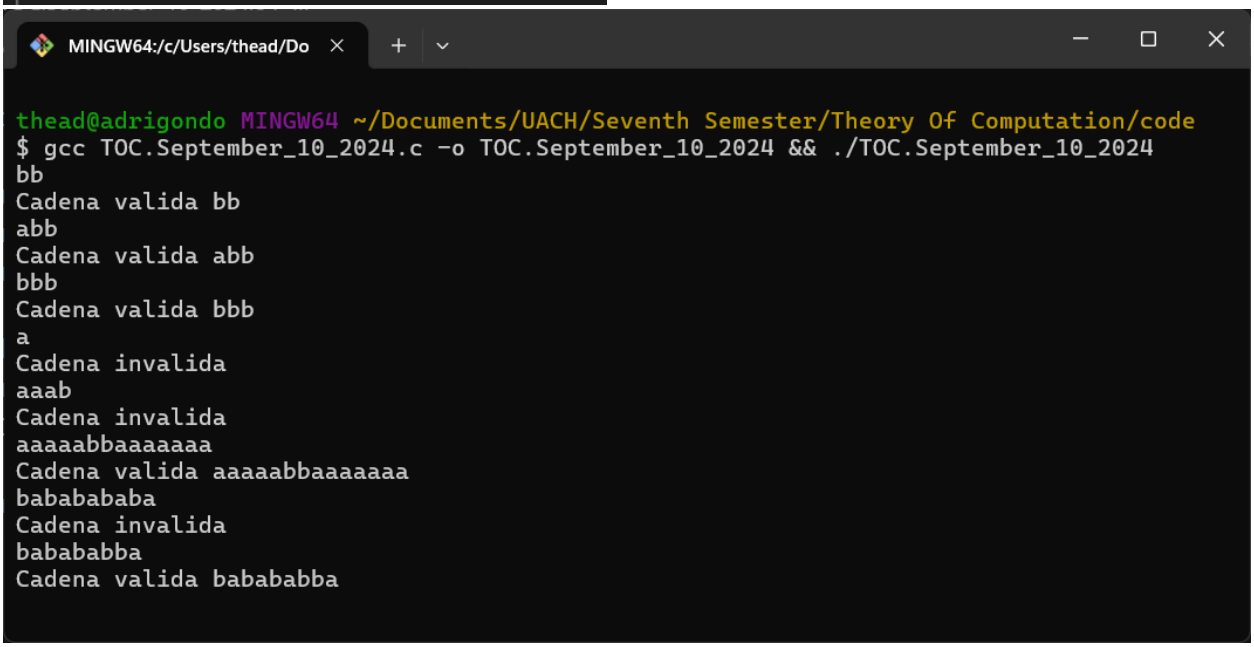
```
// Automata B
int automaton(void){
 int states[][3]={
  {2, 1, 4}, //0
  {1, 1, 4}, //1
  {2, 3, 4}, //2
  {2, 3, 4}, //3
  {4, 4, 4} //4
 };
 int finals[]={1, 3};
 int finals_length=2;
 char alphabet[]={'a', 'b'};
 int alphabet_length=2;
 int state=0;
 char c;
 while(
  !feof(input_file) &&
  (c=getc(input_file))!='\n'
 ){
  result[result_index++]=c;
  int column=choose_column(
   alphabet_length, alphabet, c
  );
  state=states[state][column];
 }
 return is_final(
  finals_length, finals, state
 );
}
```

```
int choose_column(int alphabet_length, char alphabet[], char c){
  int i=0;
  while(i<alphabet_length){
    if(alphabet[i]==c) return i;
    i++;
  }
  return i;
}

int is_final(int finals_length, int finals[], int state){
  int i=0;
  while(i<finals_length){
    if(finals[i]==state) return 1;
    i++
  }
  return 0;
}
```

Para el automata A

```c
int automaton(void)
{
    int states[][3] = {
        {0, 1, 3}, // 0
        {0, 2, 3}, // 1
        {2, 2, 3}, // 2
        {3, 3, 3}, // 3
    };
    int finals[] = {2};
    int finals_length = 1;
```



Para el automata B

```c
int automaton(void)
{
    int states[][3] = {
        {2, 1, 4}, // 0
        {1, 1, 4}, // 1
        {2, 3, 4}, // 2
        {2, 3, 4}, // 3
        {4, 4, 4}, // 4
    };
    int finals[] = {1,3};
    int finals_length = 2;
```