

# **UNIVERSIDAD POLITECNICA DE YUCATAN**

## **ESTRUCTURA DE LA PROGRAMACION**

### **PROYECTO**

**Students:**

**Javier pacheco tun**

**Josue Francisco Gonzalez Calderon**

**Brayan Yahve Fonseca**

**Jorge adiel pool avila**

```

1.  int main(){
2.      FILE *pA; char cad [1000];
3.      gets (cad); //primera linea - cabecera
4.      //se crea el archivo
5.      pA=fopen("clase.txt","wb");//abre en modo escritura (bin)
6.      if (pA==NULL){ //verifica que el archivo se creo correctamente
7.          printf("\nError al creador el archivo clase.txt");
8.          return 0;
9.
10.     }
11.     fwrite(cad,1, sizeof(cad),pA);
12.     //se leen los datos de los alumnos
13.     Alumno al;
14.     scand("%s", al.nombre);
15.     while(strcmp("#",al.nombre)!=0){
16.         scanf("%s",al.apellido);
17.         scanf("%s",al.facultad);
18.         scanf("%d",&al.semestre);
19.         //copiamos estos datos al archivo
20.         fwrite(&al,sizeof(al),1,pA)
21.         scanf("%s",al.nombre);
22.     }
23.     fclose(pA);
24.     return 0;
25. }

```

2.

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

```
#include <ctype.h>
```

```
#include <time.h>
```

```
char *encodecountry(char cadena[], int *count)
```

```
{
```

```
    char *new_c = malloc(1000);
```

```
    int s, r, k;
```

```
    *count = 0;
```

```
    srand(time(NULL));
```

```

s = strlen(cadena);

*new_c = cadena;

for (int i = 0; i < s; i++)
{
    new_c[i] = cadena[i];
}

do
{
    for (int i = 1; i < s; i++)
    {
        r = rand() % 10;
        if (r >= 4 && new_c[i] != '*')
        {
            new_c[i] = '*';
            *count += 1;
        }
    }
} while (*count <= 2);

//printf("%s\n", new_c);
//printf("%s\n", cadena);

return new_c;
}

```

3.

```

#include <stdio.h>

#include <string.h>

#include <stdlib.h>

```

```

//#include <time.h>

#include "encodeCountry.h"
int main()
{
    char *country = "Mexico";
    char *encodedCountry = "*****";
    int currentPoint = 0;
    srand(time(NULL));
    encodedCountry = encodecountry(country, &currentPoint); //, &currentPoint);

    printf("NameofCountry: %s \n", country);
    printf("encodeCountry: %s \n", encodedCountry);
    printf("currentPoint: %d \n", currentPoint);
    return 0;
}

```

## PARTE 2.

```

int ledPin [] = {
    9,8,7,6,5,4,3,2;
    int maxCount=256;
    int delayInterval=250;

    void setup ()
{
    //set up each pin as an output pin
    for(int i=0; i<9; i++)
    {
        pinMode(ledPin[i],output);
    }
}

```

```

    }
}
void loop ()
{
    //LOOP through each number
    for (int counter = 0; counter < maxCount; counter++)
    {
        //pass the count number to the displayBinary function
        displayBinary (counter);
        delay (delayInterval);
    }
}
void displayBinary(byte numToShow)
{
    for (int i = 0; i < 8; i++)
    {
        if(bitRead(numToShow, i)==1)
        {
            digitalWrite(ledPin[i], HIGH);
        }
        else
        {
            digitalWrite(ledPin[i], LOW)
        }
    }
}

```