

# Reporte

Ana Paola Nava Vivas

12 de Septiembre de 2016



# Capítulo 1

## Potencia de alfabeto

### 1.1. Descripción del programa 1

Se tiene un alfabeto que sólo posee los caracteres '0' y '1' y se debe hallar el conjunto de todas las cadenas posibles de longitud que va desde 0 a n, que se pueden obtener combinando estos caracteres. Si se elige n=2, se tendrá  $\Sigma = e, 0, 1, 00, 01, 10, 11$ . Además, el programa debe tener las opciones manual y automática, en donde n será un número random para ésta última.

### 1.2. Código del programa 1

A continuación se muestra el código del programa en c++.

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>

int Binario(int);

int main() {
    int op,n,m,h;

    do{
        printf("\n\n1. Escoger numero\n2. Random\n0 para salir\n");
        scanf("%d",&op);
        switch(op){
            case 1:
                scanf("%d",&n);
                FILE *ar;
                ar=fopen("Universo.txt","a");
                printf("e,");
                fprintf(ar,"\n\nModo Manual");
                fprintf(ar,"\ne,");
```

```

        fclose(ar);
        for(h=1;h<=n;h++){
            Binario(h);}

        break;
    case 2:
        m=rand()%1001;
        printf("\n%d\n\n",m);
        FILE *arch;
        arch=fopen("Universo.txt","a");
        printf("e,");
        fprintf(arch,"\n\nModo Automatico");
        fprintf(arch,"\ne,");
        fclose(arch);
        for(h=1;h<=m;h++){
            Binario(h);}
        break;
    default:
        break;
    }}
while(op!=0);
return 0;
}

```

```

int Binario(int l){
    int a[l];
    int k,i,z,p,j;
    int powah=pow(2,l)-1;
    for(k=0;k<=powah;k++){
        z=k;
        i=0;

        while(z>0){
            if(z%2==0){
                a[i]=0;
                i++;}

            else{
                a[i]=1;
                i++;
            }

            z=z/2;
        }

        FILE *arc;
        arc=fopen("Universo.txt","a");
        for(p=1;p<=l-i;p++){
            printf("0");
            fputc('0',arc);
        }
        for(j=i-1;j>=0;j--){

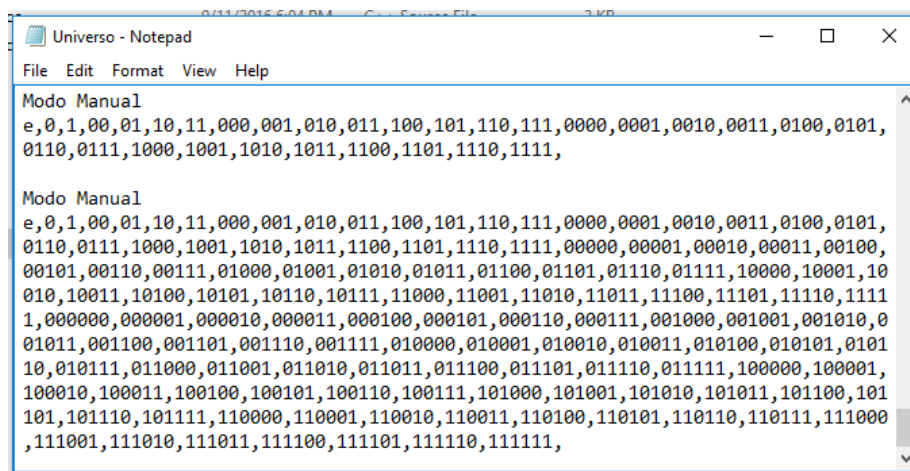
```

```

        printf("%d",a[j]);
        fprintf(arc,"%d",a[j]);}
printf(",");
fprintf(arc," ,");
fclose(arc);
}
}

```

### 1.3. Resultados del programa 1



```
Universo - Notepad
File Edit Format View Help

Modo Automatico
e,0,1,00,01,10,11,000,001,010,011,100,101,110,111,0000,0001,0010,0011,0100,0101,
0110,0111,1000,1001,1010,1011,1100,1101,1110,1111,00000,00001,00010,00011,00100,
00101,00110,00111,01000,01001,01010,01011,01100,01101,01110,01111,10000,10001,10
010,10011,10100,10101,10110,10111,11000,11001,11010,11011,11100,11101,11110,1111
1,000000,000001,000010,000011,000100,000101,000110,000111,001000,001001,001010,0
01011,001100,001101,001110,001111,010000,010001,010010,010011,010100,010101,0101
10,010111,011000,011001,011010,011011,011100,011101,011110,011111,100000,100001,
100010,100011,100100,100101,100110,100111,101000,101001,101010,101011,101100,101
101,101110,101111,110000,110001,110010,110011,110100,110101,110110,110111,111000
,111001,111010,111011,111100,111101,111110,111111,

Modo Automatico
e,0,1,

Modo Automatico
e,0,1,00,01,10,11,000,001,010,011,100,101,110,111,0000,0001,0010,0011,0100,0101,
0110,0111,1000,1001,1010,1011,1100,1101,1110,1111,00000,00001,00010,00011,00100,
00101,00110,00111,01000,01001,01010,01011,01100,01101,01110,01111,10000,10001,10
010,10011,10100,10101,10110,10111,11000,11001,11010,11011,11100,11101,11110,1111
1,000000,000001,000010,000011,000100,000101,000110,000111,001000,001001,001010,0
01011,001100,001101,001110,001111,010000,010001,010010,010011,010100,010101,0101
10,010111,011000,011001,011010,011011,011100,011101,011110,011111,100000,100001,
100010,100011,100100,100101,100110,100111,101000,101001,101010,101011,101100,101
101,101110,101111,110000,110001,110010,110011,110100,110101,110110,110111,111000
,111001,111010,111011,111100,111101,111110,111111,

Modo Automatico
e,0,1,00,01,10,11,000,001,010,011,100,101,110,111,0000,0001,0010,0011,0100,0101,
0110,0111,1000,1001,1010,1011,1100,1101,1110,1111,00000,00001,00010,00011,00100,
00101,00110,00111,01000,01001,01010,01011,01100,01101,01110,01111,10000,10001,10
010,10011,10100,10101,10110,10111,11000,11001,11010,11011,11100,11101,11110,1111
1,|

Modo Automatico
e,0,1,00,01,10,11,000,001,010,011,100,101,110,111,
```

## Capítulo 2

# Primos en Binario

### 2.1. Descripción del segundo programa

El programa calcula si un número dado es primo o no, en el caso de serlo, imprime todos los primos desde 2 hasta el número dado, primero en decimal y luego en binario. Adicional a este programa, se realizó una gráfica en Excel que sirve para ver las ocurrencias de 1's y 0's de todos los primos en un rango de 2 a 1000.

### 2.2. Código del segundo programa

A continuación el código en c++.

```
#include <stdio.h>
#include <stdlib.h>

int  Binario(int );
int  Cousin(int );

int  main() {
    system("color 01");
    system("color F0");

    int  op,n,m,k,r;

do{
    printf("\n\n1. Escoger numero\n2. Random: ");
    scanf("%d",&op);
    switch(op){
        case 1:
            FILE *ar;
```

```

ar=fopen("Primos.txt","a");
scanf("%d",&n);
fprintf(ar,"\n\nModo Manual:  %d\n",n);
    for(k=0;k<=n;k++){
        if(Cousin(k)==1){
            printf("%d",k);
            fprintf(ar,"%d, ",k);}}
printf("\n\n");
fprintf(ar,"\n\n");
fclose(ar);
    for(k=0;k<=n;k++){
        if(Cousin(k)==1){
            Binario(k);}}
break;

case 2:
FILE *arc;
arc=fopen("Primos.txt","a");
m=rand()%1001;
fprintf(arc,"\n\nModo Automatico:  %d\n",m);
printf("El numero random es:  %d\n\n",m);
    for(r=0;r<=m;r++){
        if(Cousin(r)==1){
            printf("%d",r);
            fprintf(arc,"%d, ",r);}}
printf("\n\n");
fprintf(arc,"\n\n");
fclose(arc);
    for(r=0;r<=m;r++){
        if(Cousin(r)==1){
            Binario(r);}}
break;
}}while(op==1||op==2);
return 0;
}

```

```

int Binario(int n){
FILE *arch;
arch=fopen("Primos.txt","a");

    int i=0;
    int j;
    int a[10];

while(n>0){
    if(n%2==0){
        a[i]=0;
        i++;}
    else{

```



```

        a[i]=1;
        i++;
    }
    n=n/2;}

for (j=i-1;j >=0;j--){
    printf("%d",a[j]);
    fprintf(arch,"%d",a[j]);
}
printf(",");
fprintf(arch,",");
fclose(arch);}

int Cousin(int o){
    int w;
    int cont2=0;
    if (o==1||o==0){
        return 0;}

    else{
        for (w=1;w<o;w++){
            if (o%w==0){
                cont2++;}
        }
        if (cont2>1)
            return 0;
        else
            return 1;
    }
}

```

## 2.3. Resultados del programa 2

```

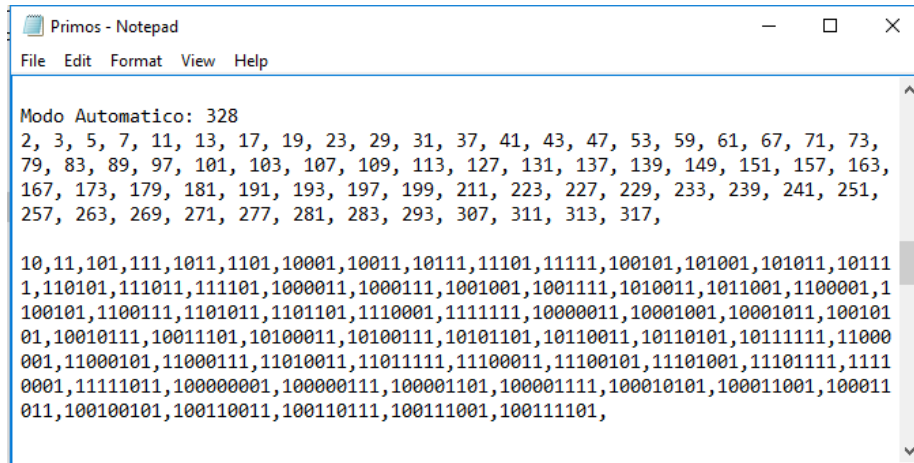
Primos - Notepad
File Edit Format View Help

Modo Manual: 70
2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67,

10,11,101,111,1011,1101,10001,10011,10111,11101,11111,100101,101001,101011,10111
1,110101,111011,111101,1000011,

Modo Manual: 400
2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73,
79, 83, 89, 97, 101, 103, 107, 109, 113, 127, 131, 137, 139, 149, 151, 157, 163,
167, 173, 179, 181, 191, 193, 197, 199, 211, 223, 227, 229, 233, 239, 241, 251,
257, 263, 269, 271, 277, 281, 283, 293, 307, 311, 313, 317, 331, 337, 347, 349,
353, 359, 367, 373, 379, 383, 389, 397,

```



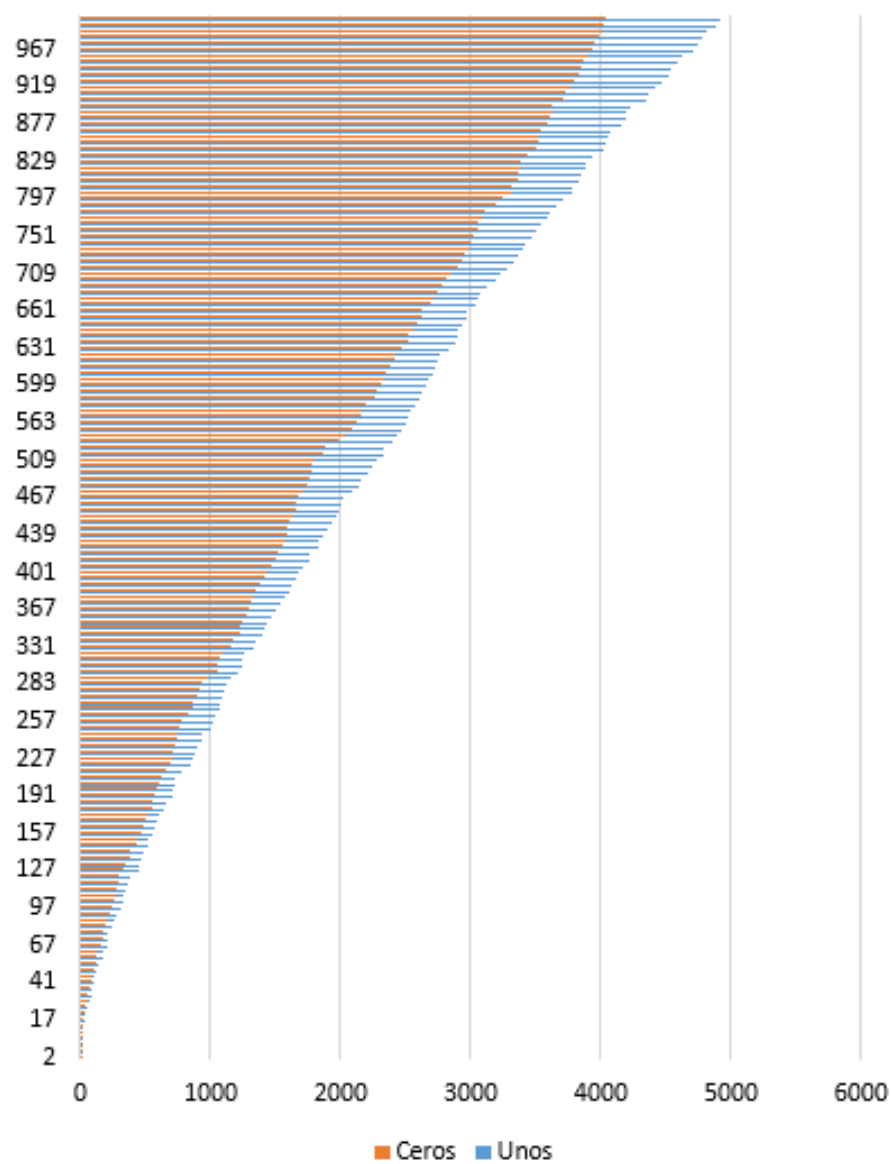
```
Primos - Notepad
File Edit Format View Help

Modo Automatico: 328
2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61, 67, 71, 73,
79, 83, 89, 97, 101, 103, 107, 109, 113, 127, 131, 137, 139, 149, 151, 157, 163,
167, 173, 179, 181, 191, 193, 197, 199, 211, 223, 227, 229, 233, 239, 241, 251,
257, 263, 269, 271, 277, 281, 283, 293, 307, 311, 313, 317,

10,11,101,111,1011,1101,10001,10011,10111,11101,11111,100101,101001,101011,10111
1,110101,111011,111101,1000011,1000111,1001001,1001111,1010011,1011001,1100001,1
100101,1100111,1101011,1101101,1110001,1111111,10000011,10001001,10001011,100101
01,10010111,10011101,10100011,10100111,10101101,10110011,10110101,10111111,11000
001,11000101,11000111,11010011,11011111,11100011,11100101,11101001,11101111,1111
0001,1111011,10000001,10000011,10000101,10000111,100010101,100011001,100011
011,100100101,100110011,100110111,100111001,100111101,
```

## 2.4. Ocurrencias de ceros y unos

Adicional al programa que encuentra números primos en un rango de 0 a 1000, se realizó otro programa que manda a un archivo todos los ceros y unos de todos los números anteriores al primo y el primo en binario, este archivo .txt se envió al programa de Excel y se realizó una gráfica para analizar estas cantidades.



## Capítulo 3

# Terminaciones en Ere

### 3.1. Descripción del programa 3

El programa consiste en un autómata que lee una cadena ya sea porque la introdujo el usuario, o porque estaba previamente guardada en un archivo 'txt'. El autómata debe encontrar palabras que terminen en 'ere', determinar cuántas son, su ubicación en el texto e imprimirlas.

### 3.2. Código del tercer programa

A continuación se mostrará el código del tercer programa en python. Es de hacer notar que este programa no está terminado, es sólo un borrador.

```
archivo=open('mesta.txt','a+')
estado=0
columna=0
linea=1
n=3
while n != 0:
    n=int(input("introduzca opcion, 1-Manual, 2-Leer archivo, otro para salir"))
    if n==1:
        cad=input("ingrese cadena: ")
        for k in cad:
            columna+=1
            if (k=='e' or k=='é' or k=='è' or k=='E' or k=='È' or k=='É' or k=='È' or k=='É'):
                if(estado==0):
                    estado=1
                elif(estado==2):
                    estado=3
            else:
                estado=1
        elif (k=='r' or k=='R'):
            if(estado==1):
```

```

        estado=2
    elif (estado==3):
        estado=2
    else:
        estado=0;

    if k=='\n':
        columna=0
        linea+=1
    if (estado==3):
        print ("q"+str(estado)+" en linea: "+str(linea)+"
Columna: "+str(columna))
    elif n==2:
        for line in archivo:
            for i in line:
                columna+=1
                if (i=='e' or i=='é' or i=='è' or i=='E' or i=='Ë' or i=='É' or i==
                    if (estado==0):
                        estado=1
                    elif (estado==2):
                        estado=3
                else:
                    estado=1
            elif (i=='r' or i=='R'):
                if (estado==1):
                    estado=2
                elif (estado==3):
                    estado=2
        else:
            estado=0;

    if i=='\n':
        columna=0
        linea+=1
    if (estado==3):
        print ("q"+str(estado)+" en linea: "+str(linea)+"
Columna: "+str(columna))#casteo
        #imprimir solo estado... la linea y columna sera para las palab

    else:
        print("opcion invalida , abortar...")
        n=0
print("se acabo")

```

## Capítulo 4

# Paridad

### 4.1. Descripción del programa 4

El programa consiste en un autómata que puede leer una cadena donde sólo hay ceros y unos, y debe saber si hay cantidad par de estos. Debe tener una opción manual donde el usuario introducirá la cadena de ceros y unos que escoja, y debe tener la opción aleatoria en donde se generará una cadena de longitud aleatoria entre 0 y 1000 caracteres de ceros y unos aleatorios.

### 4.2. Código del cuarto programa

A continuación se muestra el código del cuarto programa en python.

```
from random import randint
f=open('Paridad.txt','a')
estado=0
from tkinter import *

def Graficar():
    ventana = Tk()
    c = Canvas(ventana,width=500,height=500)
    ventana.geometry("500x500")

    c.create_oval(33,30,320,317,fill="Cyan")
    c.create_oval(48,45,305,302,fill="White")
    c.create_oval(41,41,126,126,fill="pink")####
    c.create_oval(56,56,109.5,109.5,fill="pink")
    c.create_oval(226,226,311,311,fill="pink")#abaj dere
    c.create_oval(41,226,126,311,fill="pink")#abaj izq
    c.create_oval(226,41,311,126,fill="pink")#arri dere

    c.create_oval(44,104,54,115,fill="black")
    c.create_oval(55,226,66,237,fill="black")
    c.create_oval(300,104,310,115,fill="black")
```

```

c.create_oval(285,236,296,225,fill="black")
c.create_oval(224,296,235,285,fill="black")
c.create_oval(108,306,119,295,fill="black")
c.create_oval(103,51,114,40,fill="black")
c.create_oval(226,64,237,53,fill="black")
estado0=Label(ventana, text="q0",bg="pink",font="Helvetica 16").place(x=68,y=68)
estado1=Label(ventana, text="q1",bg="pink",font="Helvetica 16").place(x=255,y=68)
estado2=Label(ventana, text="q2",bg="pink",font="Helvetica 16").place(x=68,y=255)
estado3=Label(ventana, text="q3",bg="pink",font="Helvetica 16").place(x=255,y=255)
cero1=Label(ventana, text="0",bg="white",font="Helvetica 12").place(x=288,y=170)
cero2=Label(ventana, text="0",font="Helvetica 12").place(x=323,y=170)
cero3=Label(ventana, text="0",bg="white",font="Helvetica 12").place(x=50,y=170)
cero4=Label(ventana, text="0",font="Helvetica 12").place(x=15,y=170)
uno1=Label(ventana, text="1",bg="white",font="Helvetica 12").place(x=167,y=47)
uno2=Label(ventana, text="1",font="Helvetica 12").place(x=167,y=6)
uno3=Label(ventana, text="1",bg="white",font="Helvetica 12").place(x=167,y=277)
uno4=Label(ventana, text="1",font="Helvetica 12").place(x=167,y=320)
c.create_line(10,80,41,80,fill="black")
c.create_oval(38,77,43,84,fill="black")
inicio=Label(ventana, text="inicio",font="Helvetica 10").place(x=9,y=46)

c.place(x=0,y=0)
ventana.mainloop()

```

```

c=input("Introducir m(manual), a(automatico), g(ver grafico), s(salir): ")

```

```

while(c!='s'):

```

```

    if(c=='m'):
        cad=input("introduce la cadena: ")
        f=open('Paridad.txt','a')
        f.write("\n\nMODULO MANUAL\n")
        f.write("-q0-\n")
        for i in cad:
            f.write(i+" ")
            if(i=='1'):
                if(estado==0):
                    estado=1
                    print("q"+str(estado))
                    f=open('Paridad.txt','a')
                    f.write("q"+str(estado)+"\n")
                elif(estado==1):
                    estado=0
                    print("q"+str(estado))
                    f.write("q"+str(estado)+"\n")
                elif(estado==3):
                    estado=2
                    print("q"+str(estado))

```

```

        f.write("q"+str(estado)+"\n")
    else:
        estado=3
        print("q"+str(estado))
        f.write("q"+str(estado)+"\n")
elif(i=='0'):
    if(estado==0):
        estado=2
        print("q"+str(estado))
        f.write("q"+str(estado)+"\n")
    elif(estado==2):
        estado=0
        print("q"+str(estado))
        f.write("q"+str(estado)+"\n")
    elif(estado==1):
        estado=3
        print("q"+str(estado))
        f.write("q"+str(estado)+"\n")
    else:
        estado=1
        print("q"+str(estado))
        f.write("q"+str(estado)+"\n")
else:
    print("invalido")
if(int(estado)==0): #en if estado se castea con int... en print con
    estado=0
    print("Hay paridad de ceros y unos")
    f.write("\nHay paridad de ceros y unos\n")
    c=input("Introducir m(manual), a(automatico), g(ver grafico), s")

else:
    estado=0
    print("NO hay paridad de ceros y unos")
    f.write("\nNo hay paridad de ceros y unos\n")

    c=input("Introducir m(manual), a(automatico), g(ver grafico), s")

elif(c=='a'):
    long=randint(0,1000)#
    print("longitud: "+str(long))
    f=open('Paridad.txt','a')
    f.write("\n\nMODULO AUTOMATICO\n")
    f.write("\nlongitud: "+str(long))
    cad3=""
    for h in range(0,long):#
        cad2=str(randint(0,1))
        cad3=str(cad3+cad2)
    print(""+str(cad3))
    print("q0, ")
    f.write("\n"+str(cad3)+"\n")

```



```

f.write("\nq0, ")
for i in cad3:
    if(i=='1'):
        if(estado==0):
            estado=1
            print("q"+str(estado))
            f.write(" q"+str(estado)+" = 1,")
        elif(estado==1):
            estado=0
            print("q"+str(estado))
            f.write(" q"+str(estado)+" = 1,")
        elif(estado==3):
            estado=2
            print("q"+str(estado))
            f.write(" q"+str(estado)+" = 1,")
        else:
            estado=3
            print("q"+str(estado))
            f.write(" q"+str(estado)+" = 1,")
    elif(i=='0'):
        if(estado==0):
            estado=2
            print("q"+str(estado))
            f.write(" q"+str(estado)+" = 0,")
        elif(estado==2):
            estado=0
            print("q"+str(estado))
            f.write(" q"+str(estado)+" = 0,")
        elif(estado==1):
            estado=3
            print("q"+str(estado))
            f.write(" q"+str(estado)+" = 0,")
        else:
            estado=1
            print("q"+str(estado))
            f.write(" q"+str(estado)+" = 0, ")
    else:
        print("invalido")
if(int(estado)==0):
    estado=0
    print("Hay paridad de ceros y unos")
    f.write("\nHay paridad de ceros y unos\n")
    f.close()
    c=input("Introducir m(manual), a(automatico), g(ver grafico) s(salir):")
else:
    estado=0
    print("NO hay paridad de ceros y unos")
    f.write("\nNo hay paridad de ceros y unos\n")
    f.close()
    c=input("Introducir m(manual), a(automatico), g(ver grafico), s(salir):")

```

```

elif (c=='g'):
    Graficar()
    c=input(" Introducir m(manual), a(automatico), g(ver grafico), s(sa

else:
    print("\nopcion no valida")
    c=input(" Introducir m(manual), a(automatico), g(ver grafico), s(sa

```

### 4.3. Resultados del programa 4

```

MODULO AUTOMATICO

longitud: 101
1011010111000001000111101110000011010110010100110001000101111011
1001110101011110000010110010101101101

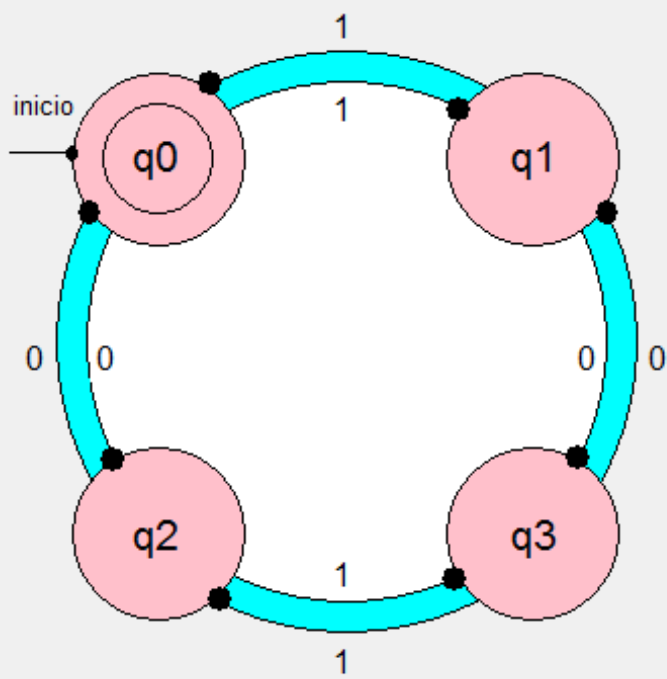
q0, q1 = 1, q3 = 0, q2 = 1, q3 = 1, q1 = 0, q0 = 1, q2 = 0, q3
= 1, q2 = 1, q3 = 1, q1 = 0, q3 = 0, q1 = 0, q3 = 0, q1 = 0,
q0 = 1, q2 = 0, q0 = 0, q2 = 0, q3 = 1, q2 = 1, q3 = 1, q2 = 1,
q0 = 0, q1 = 1, q0 = 1, q1 = 1, q3 = 0, q1 = 0, q3 = 0, q1 = 0,
q3 = 0, q2 = 1, q3 = 1, q1 = 0, q0 = 1, q2 = 0, q3 = 1, q2 =
1, q0 = 0, q2 = 0, q3 = 1, q1 = 0, q0 = 1, q2 = 0, q0 = 0, q1 =
1, q0 = 1, q2 = 0, q0 = 0, q2 = 0, q3 = 1, q1 = 0, q3 = 0, q1 =
0, q0 = 1, q2 = 0, q3 = 1, q2 = 1, q3 = 1, q2 = 1, q0 = 0, q1 =
1, q0 = 1, q1 = 1, q3 = 0, q1 = 0, q0 = 1, q1 = 1, q0 = 1, q2 =
0, q3 = 1, q1 = 0, q0 = 1, q2 = 0, q3 = 1, q2 = 1, q3 = 1, q2 =
1, q0 = 0, q2 = 0, q0 = 0, q2 = 0, q0 = 0, q1 = 1, q3 = 0, q2 =
1, q3 = 1, q1 = 0, q3 = 0, q2 = 1, q0 = 0, q1 = 1, q3 = 0, q2 =
1, q3 = 1, q1 = 0, q0 = 1, q1 = 1, q3 = 0, q2 = 1,
No hay paridad de ceros y unos

MODULO MANUAL
-q0-
1 q1
0 q3
1 q2
0 q0
1

```

### 4.4. Gráfico del programa 4

tk



## Capítulo 5

# Protocolo

### 5.1. Descripción del programa 5

De manera aleatoria, el programa debe simular que está encendido ‘1’ o no ‘0’. Si está apagado se vuelve a disparar un ‘random’, si está encendido se deben generar 50 cadenas de 32 caracteres de ceros y unos aleatorios que se enviarán a una archivo .txt. El programa se pausará un segundo y luego abrirá otro archivo .txt donde se introducirán las cadenas que cuentan con paridad de ceros y unos del archivo original.

### 5.2. Código del quinto programa

A continuación se muestra el código del quinto programa en python.

```
from time import sleep #sleep(segundos)
from random import randint
from tkinter import *

def Graficar():
    ventana = Tk()
    c = Canvas(ventana, width=500, height=500)
    ventana.geometry("500x500")
    c.create_oval(48,71,305,156)
    c.create_oval(256,81,341,146)
    c.create_oval(256,81,341,146)
    c.create_oval(41,71,126,156, fill="pink")####
    c.create_oval(226,71,311,156, fill="pink")#arri dere
    c.create_line(5,110,41,110)
    c.create_oval(38,107,44,112, fill="black")
    c.create_oval(107,145,116,154, fill="black")
    c.create_oval(240,74,250,83, fill="black")
    c.create_oval(290,145,300,154, fill="black")
    start=Label(ventana, text="start", font="Helvetica 11").place(x=7,y=80)
    estado0=Label(ventana, text="ready", bg="pink", font="Helvetica 16").place
```

```

estado0=Label(ventana , text="sending" , bg="pink" , font="Helvetica 16").place(x=230,y=45)
datain=Label(ventana , text="data in" , font="Helvetica 13").place(x=150,y=45)
datain=Label(ventana , text="back" , font="Helvetica 13").place(x=153,y=157)
cadenas=Label(ventana , text="50 cadenas" , font="Helvetica 13").place(x=226,y=45)
seg=Label(ventana , text="1 seg" , font="Helvetica 13").place(x=342,y=100)
c.place(x=0,y=0)
ventana.mainloop()

c=input("Intentar(v) , grafico(g) , salir(s): ")
while(c!='s'):

    if(c=='v'):
        onOff=randint(0,1)
        if(str(onOff)=='1'):
            for i in range(0,50):# genera 50 cadenas
                cad3=""
                for h in range(0,32):#32 caracteres
                    cad2=str(randint(0,1))#ceros y 1's random
                    cad3=str(cad3+cad2)#concatenando en la cadena
                if(h==31):#para que no imprima piramedes
                    print(str(cad3))
                    f=open('Cadena.txt','a')
                    f.write(str(cad3)+"\n")
                    f.close()
            c=input("Intentar(v) , grafico(g) , salir(s): ")
            sleep(1)
            l=open('Cadena.txt','r+')
            for line in l.readlines():
                estado=0
                for k in line:
                    w=k
                    if(k=='1'):
                        if(estado==0):
                            estado=1
                        elif(estado==1):
                            estado=0
                        elif(estado==3):
                            estado=2
                    else:
                        estado=3
            elif(k=='0'):
                if(estado==0):
                    estado=2
                elif(estado==2):
                    estado=0
                elif(estado==1):
                    estado=3
            else:
                estado=1

```

```

        if (estado==0):
            ar=open(' Analisis.txt ','a+')
            for w in line:
                ar.write(str(w))
            ar.close()

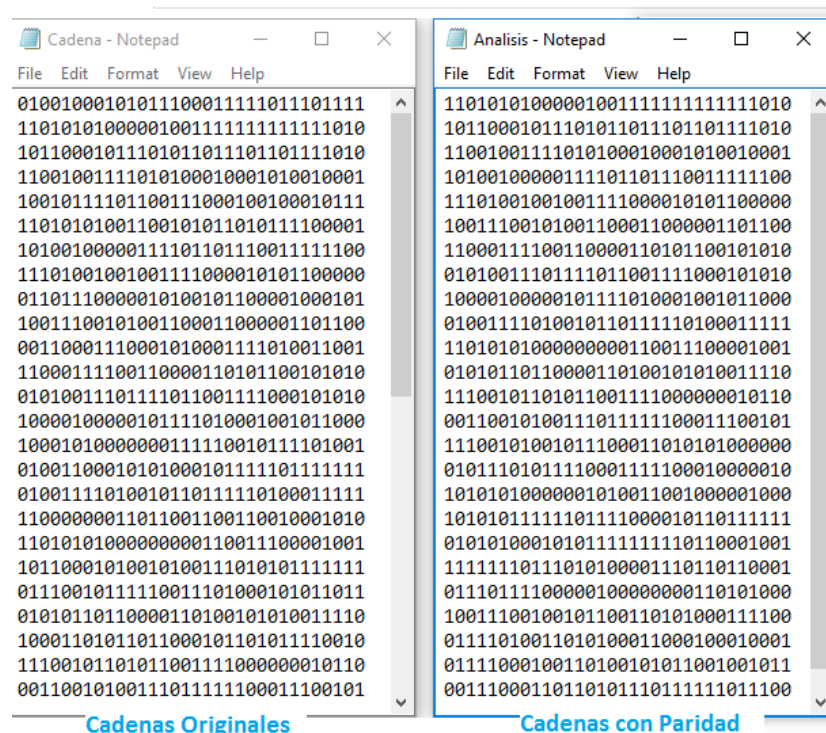
        l.close()
    else:
        print(" Apagado")
        c=input(" Intentar(v), grafico(g), salir(s): ")

elif (c=='g'):
    Graficar()
    c=input(" Intentar(v), grafico(g), salir(s): ")

else:
    print(" invalido ")
    c=input(" Intentar(v), grafico(g), salir(s): ")

```

### 5.3. Resultados del programa 5



#### 5.4. Gráfico del programa 5

