

# Prueba de Caja Blanca

***“LECTURA DE PSICOPATOLOGÍAS  
(DEPRESIÓN)”***

## REQUISITO 003 Versión 4.0

### Integrantes:

**Joel Arguello  
Cesar Looz  
Josué Villavicencio**

**Fecha 2022-01-18**

### 1. REQUISITO PLANTEADO

Historia de Usuario	
Número: REQ 002	Usuario: C.I
Nombre de la Historia: Ejecución del programa	
Prioridad: Alta	
Programador Responsable: Joel Arguello	
Descripción: <ul style="list-style-type: none"><li>El usuario llegando al menú lo único que haría sería escoger cualquiera de las opciones que dispone y si estas se ejecutan de forma correcta el usuario podrá retornar al menú para escoger otra opción o cerrar el programa</li></ul>	
Validación: <ul style="list-style-type: none"><li>Si se ejecutan las tareas que se les pide se puede retornar al menú de inicio o cerrar el programa</li></ul>	

## 2. CÓDIGO FUENTE

### 2.1. Código C

```
#include<stdio.h>
#include<stdlib.h>
#include<conio.h>
#include <string.h>
#include <windows.h>

void gesuser(void)
{
    int opc;
    FILE *fp;
    char elimu[MAX];
    char cie[MAX];
    const char* str1 = "c:/DDEPS/usuarios/";
    char ubielu[MAX];
    char ubieu[MAX];
    char cir[10];
    char newpas[90];
    char clave[50];
    do{
        system("cls");
        printf(CI"\t\t\t=====
\nANSI_COLOR_RESET);
        printf(CI"\t\t\t##
        printf("\t\t\t###ANSI_COLOR_RESET"
        printf("\t\t\t###ANSI_COLOR_RESET"
        printf("\t\t\t###ANSI_COLOR_RESET"
        printf("\t\t\t###ANSI_COLOR_RESET"
        printf("\t\t\t###ANSI_COLOR_RESET"
        printf("\t\t\t###ANSI_COLOR_RESET"
        printf(CI"\t\t\t=====
\nANSI_COLOR_RESET);
        printf("\nDigite la opción a escoger: ");
        strcat(strcpy(ubielu, str1), cie);
        fflush( stdin );
        scanf("%d", &opc);
        switch(opc){
            case 1:
                system("cls");
                ruser();
                break;
            case 2:
                system("cls");
                fflush(stdin);
                system("cls");

        printf(CI"\t\t\t=====
\nANSI_COLOR_RESET);
        printf(CI"\t\t\t##
        printf("\t\t\t###ANSI_COLOR_RESET"
        printf(CI"\t\t\t##
        printf("\t\t\t###ANSI_COLOR_RESET"

        printf(CI"\t\t\t=====
```

```

\n"ANSI_COLOR_RESET);
    printf("Ingrese el C.I del doctor: ");
    scanf("%s",cie);
    strcat(strcpy(ubielu, str1), cie);
    fp = fopen (ubielu , "r");
    if(fp!=NULL)
    {
        fgets (clave, 50, fp);
        printf("\nUsuario: %s\n",cie);
        printf("La clave es la siguiente: ");
        puts(clave);
        getch();
        menu();
        fclose(fp);}
    else
    {

        printf(CI"\t\t\t=====
\n"ANSI_COLOR_RESET);
        printf(CI"\t\t\t##                                ##\n");
        printf("\t\t\t###"ANSI_COLOR_RESET"                No existe el usuario!

CI"##\n");
        printf("\t\t\t###"ANSI_COLOR_RESET"

CI"##\n");

        printf(CI"\t\t\t=====
\n"ANSI_COLOR_RESET);
        printf("No existe el usuario");
        getch();
        fclose(fp);
        menu();}
        break;
    case 3:
        system("cls");
        char cir[10];

        printf(CI"\t\t\t=====
\n");
        printf("\t\t\t##                                ##\n");
        printf("\t\t\t###"ANSI_COLOR_RESET"                ELIMINAR USUARIO

CI"##\n");
        printf("\t\t\t###"ANSI_COLOR_RESET"

CI"##\n");

        printf("\t\t\t=====
\n"ANSI_COLOR_RESET);
        fflush(stdin);
        printf("Digite el numero de cédula del usuario a eliminar: ");
        scanf("%s",elimu);
        strcat(strcpy(ubieiu, str1), elimu);
        if(remove(ubieiu)==0){
            system("cls");

            printf(CI"\t\t\t=====
\n");
            printf("\t\t\t##                                ##\n");
            printf("\t\t\t###"ANSI_COLOR_RESET"                C.I DEL USUARIO ELIMINADO

CI"##\n");
            printf("\t\t\t###"ANSI_COLOR_RESET"                %s

```

```

"CI"##\n",cir );
printf("\t\t\t###ANSI_COLOR_RESET"
Se eliminó de manera correcta

"CI"##\n");

printf("\t\t\t===== \n
ANSI_COLOR_RESET);
    getch();
    }else{
printf("\t\t\tHubo un error vuelva a intentar.\n");
printf("\t\t\tRecuerde que no puede borrar su propio usuario!\n");
    getch();}
    break;
    case 4:
        system("cls");
        menu();
        break;
    case 5:
        printf("Gracias por confiar en nosotros!\n");
        printf("Digite enter para salir\n");
        getch();
        exit(1);
        break;
    default:
        printf("\nOpción incorrecta");
        getch();
        break;
    }
}while(opc!=6);
}

void validar(char num[]){
    for (int i = 0; i < strlen(num); i++)
    {
        if(!isdigit(num[i]) ) {
            system("cls");

            printf(CI"\t\t\t=====
\nANSI_COLOR_RESET);
            printf(CI"\t\t\t##
            ##\n");
            printf("\t\t\t###ANSI_COLOR_RESET"
            ¡SOLO SE ACEPTAN CÉDULAS!

"CI"##\n");
            printf("\t\t\t###ANSI_COLOR_RESET"
            REGRESANDO AL MENU

"CI"##\n");
            printf("\t\t\t###ANSI_COLOR_RESET"
            "CI"##\n");

            printf(CI"\t\t\t=====
\nANSI_COLOR_RESET);
            getch();
            gesuser();
            break;
        }
    }
}

void ruser(void)
{
    int cedula;
    int ced,pares,impares,total,dec=0;
    int a,b,c,d,e,f,g,h,i,j,k;

```

```

FILE *fp;
char cir[MAX];
const char* str1 = "c:/DDEPS/usuarios/";
char ubieu[MAX];
char nombre[30];
char newpas[90];
char character;
system("cls");
printf(CI"\t\t\t=====
\n"ANSI_COLOR_RESET);
printf(CI"\t\t\t##
printf("\t\t\t###ANSI_COLOR_RESET"          ##\n");
printf("\t\t\t###ANSI_COLOR_RESET"          REGISTRO DE USUARIOS          "CI"##\n");
printf(CI"\t\t\t=====
\n"ANSI_COLOR_RESET);
fflush(stdin);
printf("Digite numero de cedula: ");
scanf("%s",cir);
strcat(strcpy(ubieu, str1), cir);
validar(cir);
cedula = atoi(cir);
ced=cedula;
a=cedula/1000000000;
cedula=cedula-(a*1000000000);
b=cedula/100000000;
cedula=cedula-(b*100000000);
c=cedula/10000000;
cedula=cedula-(c*10000000);
d=cedula/1000000;
cedula=cedula-(d*1000000);
e=cedula/100000;
cedula=cedula-(e*100000);
f=cedula/10000;
cedula=cedula-(f*10000);
g=cedula/1000;
cedula=cedula-(g*1000);
h=cedula/100;
cedula=cedula-(h*100);
i=cedula/10;
cedula=cedula-(i*10);
j=cedula/1;
cedula=cedula-(j*1);
if (cedula>2400000000){
printf("wtf");
}else{
    pares=b+d+f+h;
    a=a*2;
    if (a>9){
        a=a%10+a/10;
    }
    c=c*2;
    if (c>9){
        c=c%10+c/10;
    }
    e=e*2;
    if (e>9){
        e=e%10+e/10;
    }
    g=g*2;

```

```

if (g>9){
    g=g%10+g/10;
}
i=i*2;
if (i>9){
    i=i%10+i/10;
}
impares=a+c+e+g+i;
total=pares+impares;
while (dec-total!=j && dec<total+10){
    dec=dec+10;
}
if (dec-total==j)
{
    fflush(stdin);
    fp = fopen (ubieu      , "r");
    if(fp!=NULL){
        printf("\nError");
        printf("\nYa existe este usuario");
        getch();
        fclose(fp);}
    else {
        i = 0;
        fp = fopen (ubieu      , "w+");
        printf("Digite su clave: ");
        while (caracter = getch()) {
            if (caracter == TECLA_ENTER) {
                newpas[i] = '\0';
                break;
            } else if (caracter == TECLA_BACKSPACE) {
                if (i > 0) {
                    i--;
                    printf("\b \b");
                }

            } else {
                if (i < LONGITUD) {
                    printf("*");
                    newpas[i] = caracter;
                    i++;
                }
            }
        }
    }
}

```

```

printf("a\n"ANSI_COLOR_RESET);
fputs(newpas, fp);
system("cls");

```

```

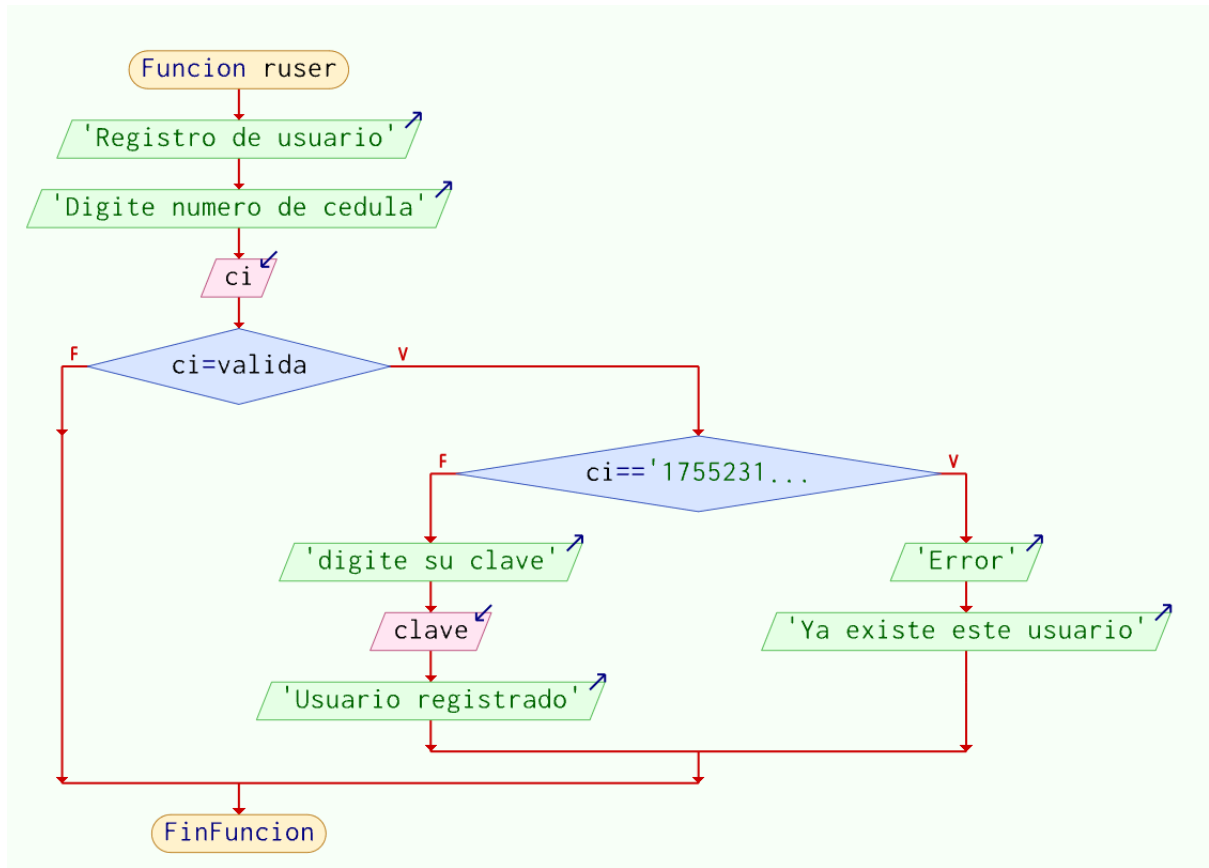
printf(CI"\n\n\n\t\t\t=====
=====\\n"ANSI_COLOR_RESET);
printf(CI"\t\t\t##
printf("\t\t\t###"ANSI_COLOR_RESET"
printf("\t\t\t###"ANSI_COLOR_RESET"
printf("\t\t\t###"ANSI_COLOR_RESET"
printf(CI"\t\t\t##

```

ÉXITO!                      "CI"##\n");  
 "CI"##\n");  
 resguardo                      "CI"##\n");

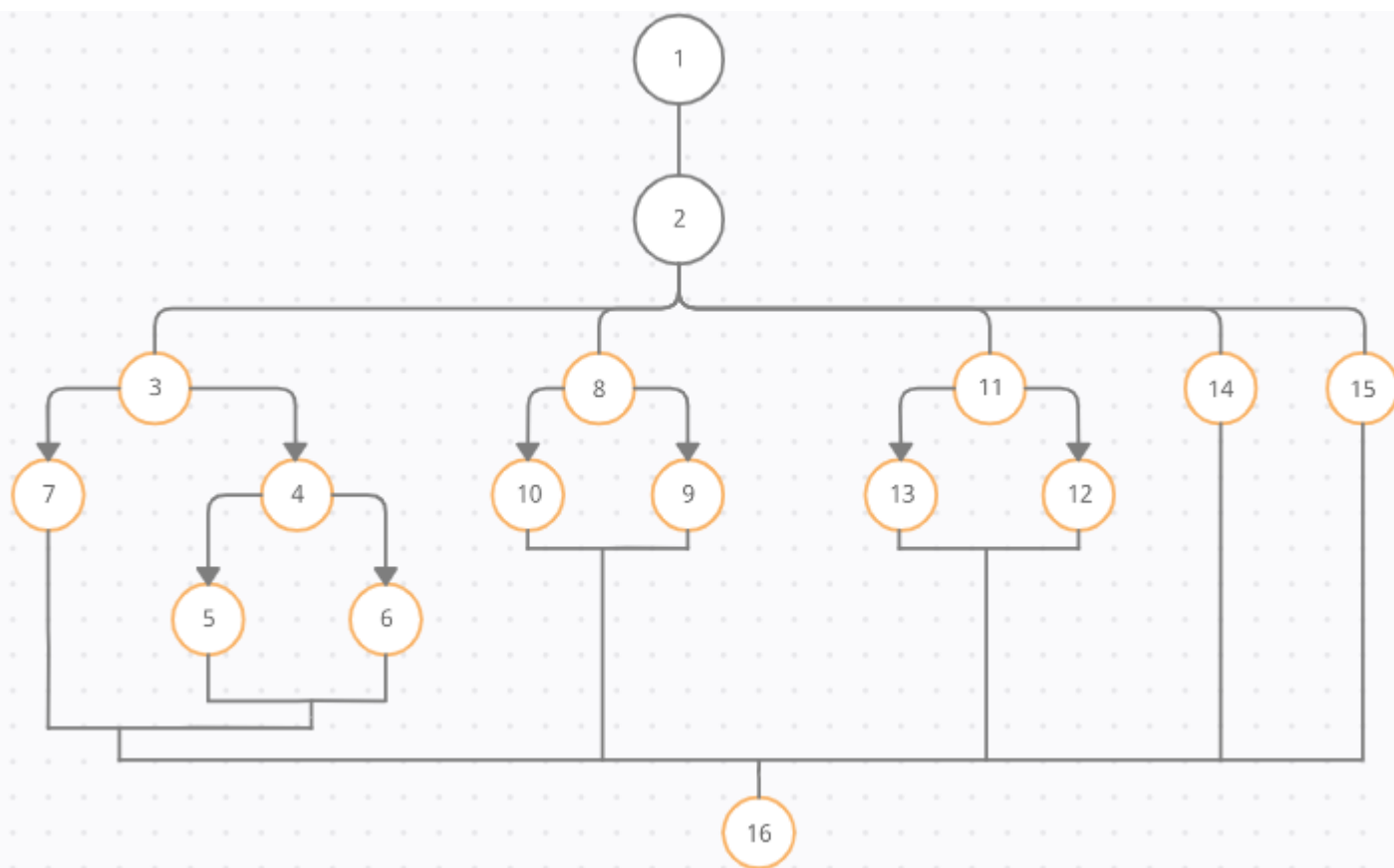
#####\n");  
 USUARIO REGISTRADO CON  
 :)  
 Recuerde tener su clave en  
 #####\n");







#### 4. GRAFO



## RUTAS

R1: 1,2,3,4,5,6,16

R2: 1,2,3,7,16

R3: 1,2,8,9,16

R4: 1,2,8,10,16

R5: 1,2,11,12,16

R6: 1,2,11,13,16

R7: 1,2,14,16

R8: 1,2,15,16

## COMPLEJIDAD CICLOMÁTICA

### EJEMPLO:

Se puede calcular de las siguientes formas:

- $V(G) = A - N + 2 = 8 - 7 + 2 = 3$

DONDE:

**A:** Número de aristas

**N:** Número de nodos

- $V(G) = \text{número de nodos predados (IF)} + 1 = 3$

- $V(G) = A - N + 2 = 29 - 16 + 2 = 15$

- $V(G) = 14 + 1 = 15$