

# Prueba de Caja Blanca

## *“LECTO DE PSICOPATOLOGÍAS (DEPRESIÓN)”*

REQUISITO 003  
**Versión 3.0**

### Integrantes:

Joel Arguello  
Cesar Loor  
Josué Villavicencio

Fecha 2022-02-13

### 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   |              |
| <b>Descripción:</b> <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> |              |
| <b>Validación:</b> <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 <iostream>
#include <string.h>
#include<locale.h>
#include <windows.h>

#define PS      "\x1b[30m"
#define CI     "\x1b[36m"
#define ANSI_COLOR_RESET   "\x1b[0m"
#define RED    "\x1b[31m"


void gesuser(void);
void ruser(void);
void registr(void);
void exm(void);
void regist(void);
void elus(void);
void exam(void);
void menu(void);
void login(void);
void vru(void);



void gesuser(void)
{ int opc;
FILE *fp;
char cir[10];
char newpas[90];
char clave[50];
do{
system("cls");
printf(CI"\t\t\t===== \n"ANSI_COLO
R_RESET);
printf(CI"\t\t\t##\n");
printf("\t\t\t###"ANSI_COLOR_RESET"          MENU           "CI"##\n");
printf("\t\t\t###"ANSI_COLOR_RESET"         [1]Registre nuevo usuario       "CI"##\n");
printf("\t\t\t###"ANSI_COLOR_RESET"         [2]Ver clave de un usuario        "CI"##\n");
printf("\t\t\t###"ANSI_COLOR_RESET"         [3]Eliminar usuario            "CI"##\n");
printf("\t\t\t###"ANSI_COLOR_RESET"         [4]Volver al menu inicial        "CI"##\n");
printf("\t\t\t###"ANSI_COLOR_RESET"         [5]Salir                  "CI"##\n");
printf("\t\t\t###"ANSI_COLOR_RESET"         "CI"##\n");
printf(CI"\t\t\t===== \n"ANSI_COLO
R_RESET);
printf("\nDigite la opcion a escoger: ");
fflush( stdin );
scanf("%d", &opc);
```

```

switch(opc){
    case 1:
        system("cls");
        ruser();
        break;
    case 2:
        system("cls");
        vru();
    case 3:
        system("cls");
        elus();
        break;
    case 4:
        system("cls");
        menu();
        break;
    case 5:
        printf("Gracias por confiar en nosotros!\n");
        printf("Digita enter para continuaur\n");
        getch();
        exit(1);
        break;
    default:
        printf("\nOpci%cn incorrecta",162);
        break;
}
}while(opc!=6);
}

```

```

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[10];
    char nombre[30];
    char newpas[90];
    printf("\t\t\tRegistro de usuarios");
    printf("\n  =====\n");
    fflush(stdin);
    printf("Digite numero de cedula: ");
    gets(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 (cir , "r");
            if(fp!=NULL){
                printf("\nError");
                printf("\nYa existe este usuario");
            }
        }
    }
}

```

```
        getch();
    }else {
        printf("cedula erronea");
        getch();
    }
}
```

```

printf("\n*Este documento es solo posible leerlo*");
fclose(fp);
getch();
}
}

void vru(void)
{
    FILE *fp;
    char cir[10];
    char clave[50];
    fflush(stdin);
    printf("Ver datos de usuario: \n");
    printf("Ingrese el C.I del doctor: ");
    gets(cir);
    fp = fopen (cir , "r");
    if(fp!=NULL)
    {
        fgets (clave, 50, fp);
        printf("\nUsuario: ");
        puts(cir);
        printf("La clave es la siguiente: ");
        puts(clave);
        getch();
        menu();
        fclose(fp);}
    else
    {
        printf("No existe el usuario");
        getch();
        fclose(fp);
        menu();}
}

void elus(void)
{
    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\t===== \n"ANSI_COLOR_RESET);
    fflush(stdin);
    printf("Digite el numero de cedula del usuario a eliminar: ");
    gets(cir);
    if(remove(cir)==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
```

```
"Cl"##\n",cir );
```

```
printf("\t\t\t=====\\nANSI_COLOR_RESET);
```

```
printf("El usuario fue eliminado correctamente.");
```

```
getch();
```

```
}
```

```
else{
```

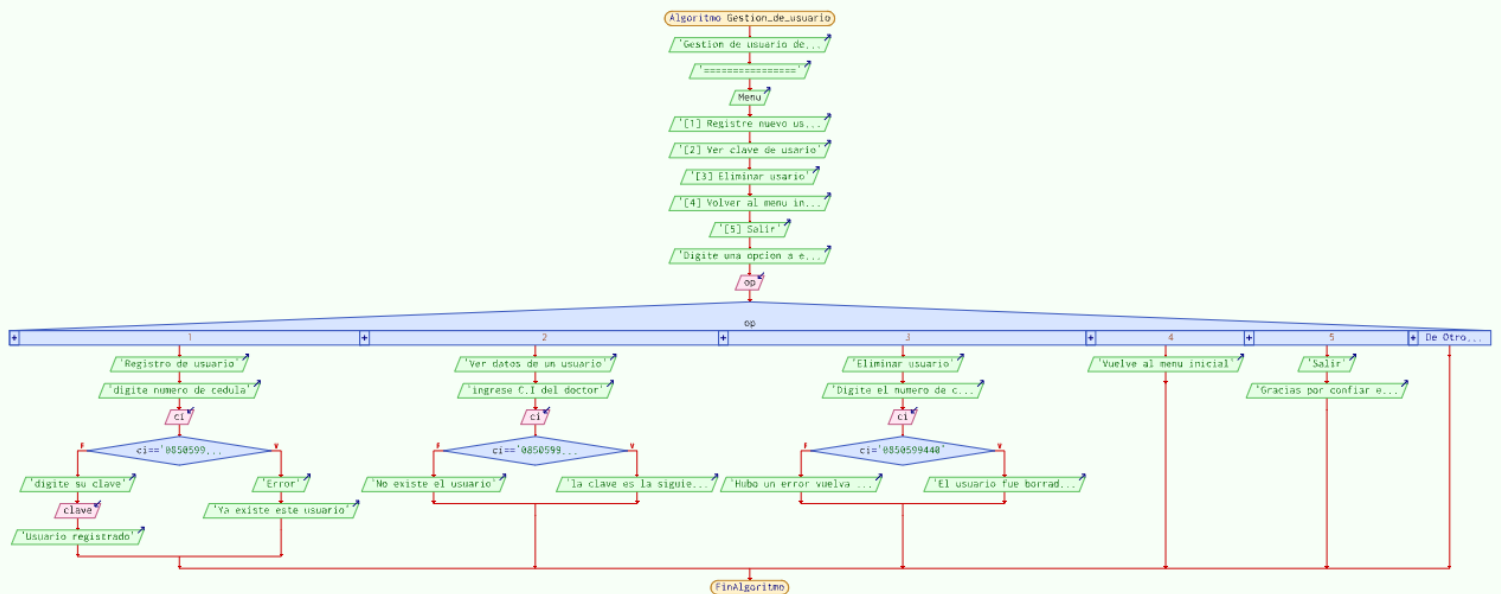
```
printf("Hubo un error vuelva a intentar.\\n");
```

```
getch();
```

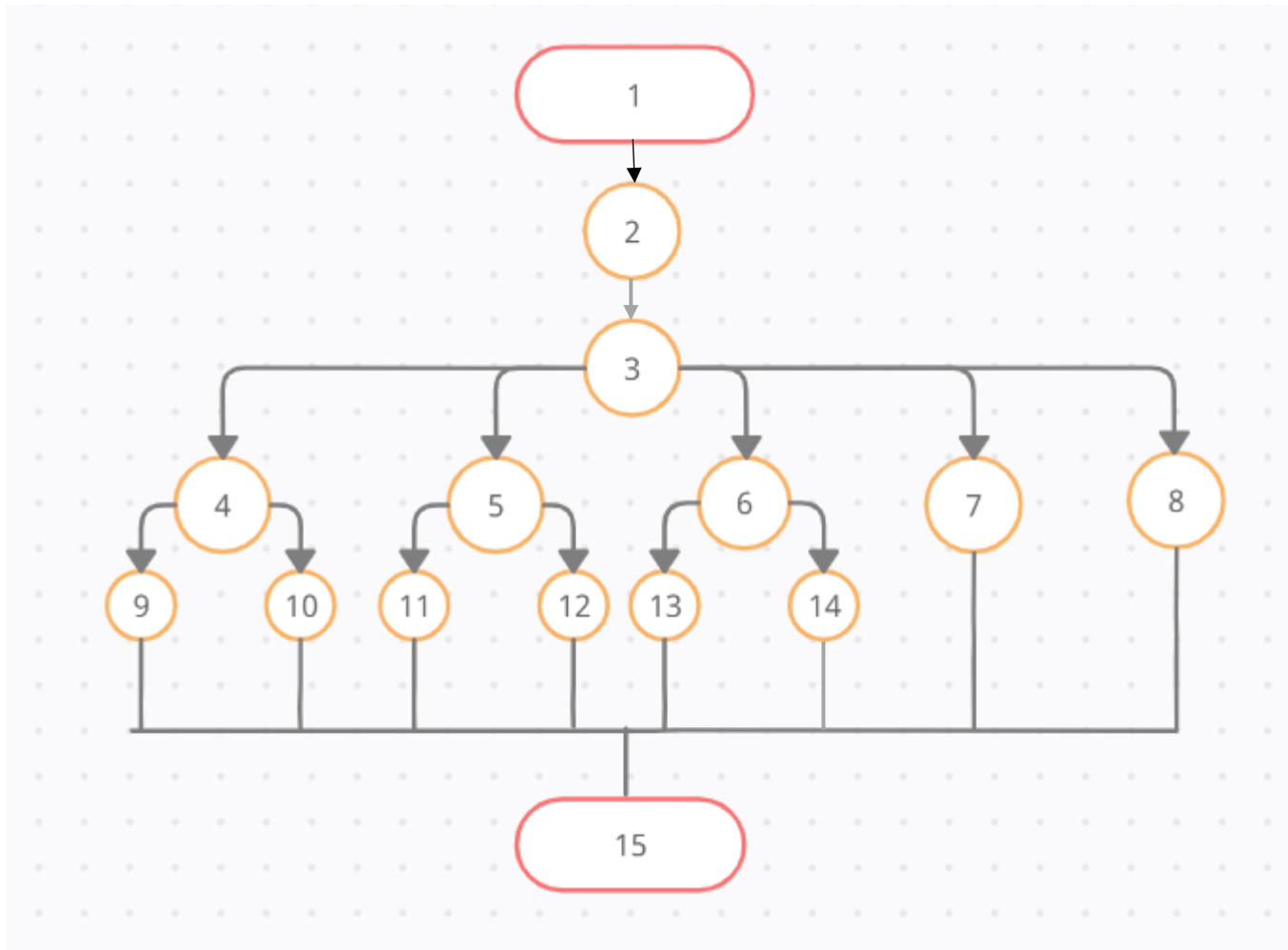
```
}
```

```
}
```

### 3. DIAGRAMA DE FLUJO



#### 4 GRAFO



#### 5. RUTAS

R1: 1,2,3,4,9,15  
R2: 1,2,3,4,10,15  
R3: 1,2,3,5,11,15  
R4: 1,2,3,5,12,15  
R5: 1,2,3,6,13,15  
R6: 1,2,3,6,14,15  
R7: 1,2,3,7,15  
R8: 1,2,3,8,15



## 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

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

$$V(G) = A - N + 2 = 25 - 15 + 2 = 12$$

$$V(G) = 11 + 1 = 12$$

