

# Prueba de Caja Blanca

---

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

## REQUISITO 002 Versión 1.0

### Integrantes:

Joel Arguello  
Cesar Loor  
Josué Villavicencio

Fecha 2022-01-18

### 1. REQUISITO PLANTEADO

Historia de Usuario	
Número: REQ 002	Usuario: C.I
Nombre de la Historia: Acceso en el sistema.	
Prioridad: Alta	
Programador Responsable: Joel Arguello	
Descripción: <ul style="list-style-type: none"><li>• Ingresar al sistema con el usuario y contraseña, desde archivo.</li></ul>	
Validación: <ul style="list-style-type: none"><li>• La interfaz muestra la página principal indica el menú principal para toda acción</li></ul>	

## 2. CÓDIGO FUENTE

### 2.1. Código C

```
1 #include<stdio.h>
2 #include<stdlib.h>
3 #include<conio.h>
4 #include <string.h>
5 #include <windows.h>
6
7 int main(){
8
9     FILE *fp;
10    char ci[10];
11    char buffer[10];
12    char contrase[25];
13    int opc = 0;
14
15    printf("\n\n\t\t\t\t\t Bienvenido a DDEOPS\n");
16    printf("\n\n\t\t\t\t\t INGRESO");
17    printf("\n\t\t\t\t\t-----");
18    login:
19    printf("\nLogin\n");
20    puts("Ingresa su C.I");
21    gets(ci);
22
23    fp = fopen (ci, "r");
24    if(fp==NULL)
25    { printf("USUARIO NO REGISTRADO\n");
26      printf("Enter para continuar.....\n");
27      goto login;
28    }
29    {
30      fgets(buffer,10,fp);
31      puts("Digite su clave para continuar");
32      gets(contrase);
33      fclose(fp);
34    }
35    if (strcmp(contrase,buffer) == 0)
36    {
37      puts("Contraseña correcta");
38      goto loginexitoso;
39    }
40    else
41    {
42      printf("Contraseña incorrecta");
43      getch();
44      exit(1);
45    }
46
47    loginexitoso:
48    {
49      system("cls");
50      printf("
51      Haz logeado correctamente");
52      system("cls");
53      goto menu;
54    }
```

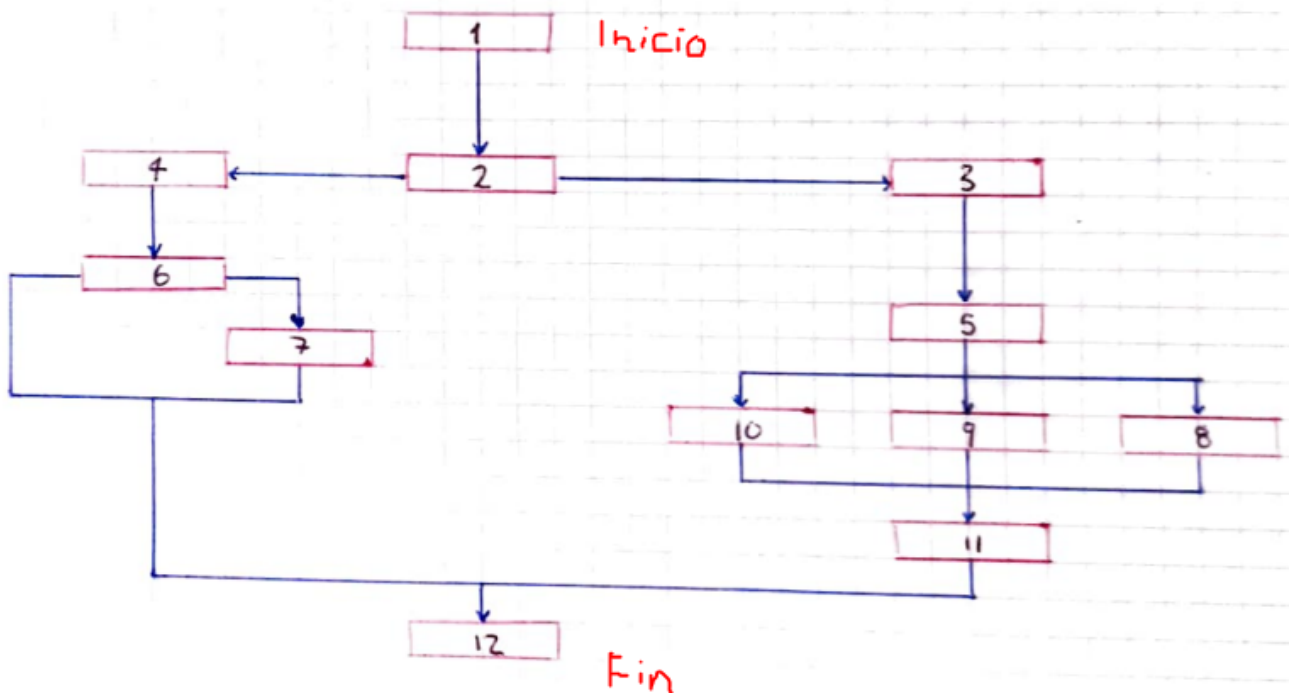
```

49     system("cls");
50     printf("                Haz logeado correctamente");
51     system("cls");
52     goto menu;
53 }
54 menu:
55 {
56     system("cls");
57     printf("\n\n\n\n\n\t\t\t\t\tMenu");
58     printf("\n\t\t\t\t\t-----\n\n\n");
59     printf("1)Ver registros\n");
60     printf("2)Realizar examen DDEDPS\n");
61     printf("3)Gestion usuarios\n");
62     printf("4)Salir\n");
63     printf("Digite opcion a escoger:\n");
64     scanf("%d",&opc);
65     switch(opc){
66     case 1:
67         system("cls");
68         goto registros;
69         break;
70     case 2:
71         system("cls");
72         goto examen;
73         break;
74     case 3:
75         system("cls");
76         goto usuarios;
77         break;
78     case 4:
79         exit(1);
80     }
81 }
82 registros:
83 {
84     printf("Registros");
85     return 0;
86 }
87 examen:
88 {
89     printf("Examen");
90     return 0;
91 }
92 usuarios:
93 {
94     printf("usuarios");
95     return 0;
96 }
97 }
98 }

```

```
graph TD
    Start([Algoritmo: Inizio]) --> DefArray[Definisci l'array: array]
    DefArray --> DefSize[Definisci la dimensione: N]
    DefSize --> SetI[Set I ← 0]
    SetI --> SetMax[array[0]]
    SetMax --> SetJ[Set J ← 1]
    SetJ --> LoopStart(( ))
    LoopStart --> ReadI[/array[J]/]
    ReadI --> Compare{array[J] > array[I]}
    Compare -- Sì --> SetMax
    Compare -- No --> SetJNext[Set J ← J + 1]
    SetJNext --> LoopStart
    SetMax --> SetINext[Set I ← I + 1]
    SetINext --> LoopStart
    LoopStart --> End([Fine])
```

#### 4. GRAFO



##### Rutas:

R1: 1,2,3,5,8,11,12

R2: 1,2,4,6,7,12

R3: 1,2,3,5,9,11,12

R4: 1,2,3,5,10,11,12

R5: 1,2,3,5,8,11,12

R6: 1,2,4,6,12

#### 5. 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

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

- $V(G) = A - N + 2 = 11 - 12 + 2 = 3$

- $V(G) = 2 + 1 = 3$