

# Prueba de Caja Blanca

---

***“SISTEMA DE GESTIÓN DE DATOS PARA UNA ORGANIZACION”***

Versión 1.2

**GRUPO N#4**

**Integrantes:**

Diego Hidalgo

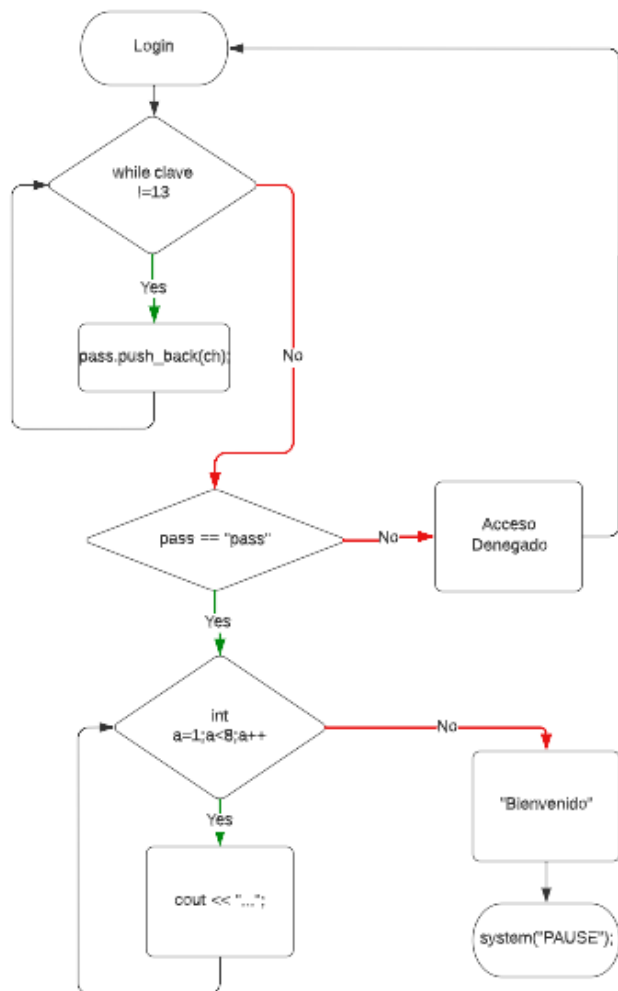
Kevin Ramos

## Prueba Caja Blanca Login

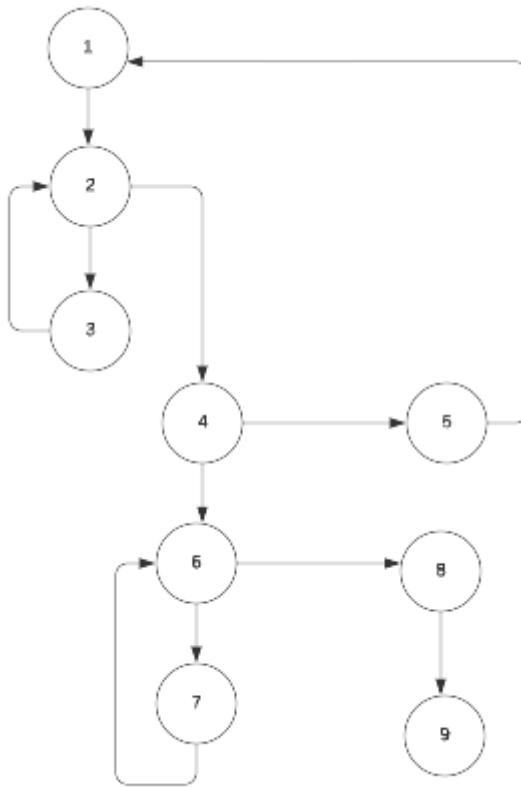
```
int login(){ //login procedure
    string pass = "";
    char ch;
    cout << "\n\n\n\n\t\t\t\t\tSISTEMA DE GESTION DE EMPLEADOS";
    cout << "\n\n\n\n\n\t\t\t\t\tIngresa tu clave admin :";
    ch = _getch();
    while(ch != 13){ //character 13 is enter
        pass.push_back(ch);
        cout << '*';
        ch = _getch();
    }
    if(pass == "pass" || pass=="amen"){
        cout<<"\n\n\n\n\t\t\t\t\tCARGANDO \n\t\t\t\t\t";
        for(int a=1;a<8;a++) // Change 'a<?' to how many * you want
        {
            Sleep(500);
            cout << "...";
        }
        cout << "\n\n\n\n\t\t\t\t\t;Acceso permitido!! \n\n\n BIENVENIDO\n\n\n";

        system("PAUSE");
        system("CLS");
    }else{
        cout << "\nAcceso cancelado o denegado...\n";
        login();
    }
}
```

### DIAGRAMA DE FLUJO



## GRAFO



### RUTAS:

R1: 1,2,3,2

R2: 1,2,3,4,5,1

R3: 1,2,3,4,6,7,6

R4: 1,2,3,4,6,7,8,9

### Complejidad Ciclomática

E: Número de aristas

N: Número de nodos

P: Número de nodos predicado

$$V(G) = E - N + 2$$

$$V(G) = 11 - 9 + 2 = 4$$

$$V(G) = P + 1$$

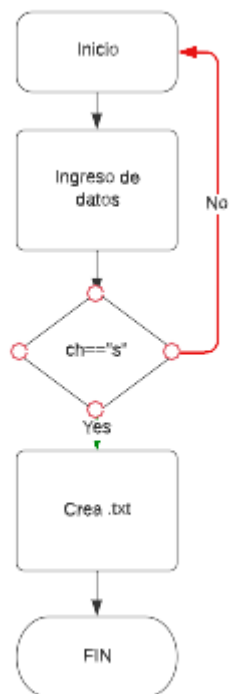
$$V(G) = 3 \text{ nodo predicado} + 1 = 4$$

## Prueba Caja Blanca Registro de Empleados

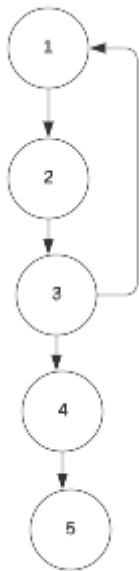
```
cin>>age;
cout<<"\n-----";
cout<<"\n Ingrese el sueldo del empleado: ";
cin>>ctc;
cout<<"\n-----";
cout<<"\n Ingrese la experiencia del empleado [Años]: ";
cin>>experience;
cout<<"\n-----";

char ch;
cout<<"\nIngrese 'S' para guardar la información anterior\n";
cin>>ch;
if(ch=='S'){
    FILE *file;
    file= fopen("data.txt","a");
    fprintf(file, "%s %s %s %d %d %d \n", name, id, designation, age, ctc, experience );
    fclose(file);
    cout<<"\nSe ha agregado un nuevo empleado a la base de datos.\n";
}
else
    addNewEmployee();
waitForEnter();
```

### DIAGRAMA DE FLUJO



### GRAFO



**RUTAS:**

R1: 1,2,3,1

R2:1,2,3,4,5

**Complejidad Ciclomática**

E: Número de aristas

N: Número de nodos

P: Número de nodos predicado

$$V(G) = E - N + 2$$

$$V(G) = 5 - 5 + 2 = 2$$

$$V(G) = P + 1$$

$$V(G) = 1 \text{ nodo predicado} + 1 = 2$$

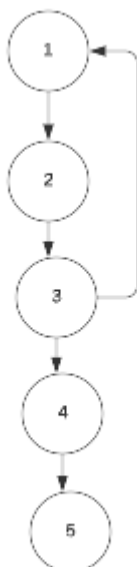
## Prueba Caja Blanca Mostrar Lista de Empleados

```
// Functions to perform desired actions
void listEmployees(void){ //To list total employees with Name, Id and Designation
    system("cls");
    FILE *file;
    file= fopen("data.txt", "r");
    cout<<"\n\t\t\t\t\tLista de empleados\n";
    cout<<"\n=====";
    cout<<"\n NOMBRE\t\t\t\t\tID\t\t\t\t\tDESIGNACION\n";
    cout<<"\n=====";
    while(fscanf(file, "%s %s %s %d %d %d", &name[0], &id[0], &designation[0], &age, &ctc, &experie
        cout<<"\n"<<name<<"\t\t"<<id<<"\t\t"<<designation;
    fclose(file);
    waitforEnter();
}
```

### DIAGRAMA DE FLUJO



### GRAFO



## RUTAS:

R1: 1,2,3,1

R2:1,2,3,4,5

## Complejidad Ciclomática

E: Número de aristas

N: Número de nodos

P: Número de nodos predicado

$$V(G) = E - N + 2$$

$$V(G) = 5 - 5 + 2 = 2$$

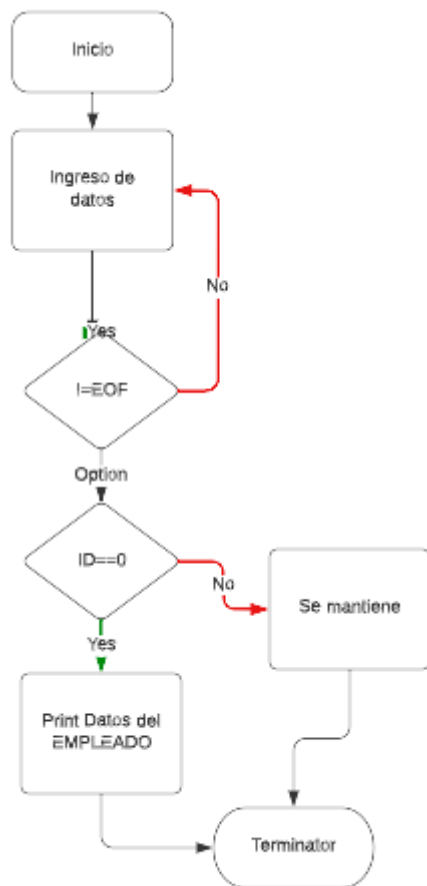
$$V(G) = P + 1$$

$$V(G) = 1 \text{ nodo predicado} + 1 = 2$$

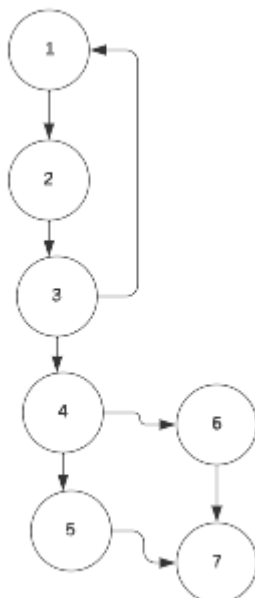
## Prueba Caja Blanca Modificar Lista de Empleados

```
char newDesignation[30];
cout<<"\n-----";
cout<<"\nIntroduzca una nueva designación: ";
cin>>newDesignation;
int newCtc;
cout<<"-----";
cout<<"\nIntroduzca un nuevo sueldo: ";
cin>>newCtc;
FILE *file, *tempfile;
file= fopen("data.txt", "r");
tempfile= fopen("temp.txt", "w");
while(fscanf(file, "%s %s %s %d %d %d", &name[0], &id[0], &designation[0], &age, &ctc, &experience) != 0)
{
    if(strcmp(checkId, id) == 0)
        fprintf(tempfile, "%s %s %s %d %d %d \n", name, id, newDesignation, age, newCtc, experience);
    else
        fprintf(tempfile, "%s %s %s %d %d %d \n", name, id, designation, age, ctc, experience);
}
fclose(file);
fclose(tempfile);
int isRemoved= remove("data.txt");
int isRenamed= rename("temp.txt", "data.txt");
waitForEnter();
}
```

## DIAGRAMA DE FLUJO



## GRAFO



**RUTAS:**

**R1: 1,2,3,2**



R2: 1,2,3,4,5,7

R3:1,2,3,4,6,7

### **Complejidad Ciclomática**

E: Número de aristas

N: Número de nodos

P: Número de nodos predicado

$$V(G) = E - N + 2$$

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

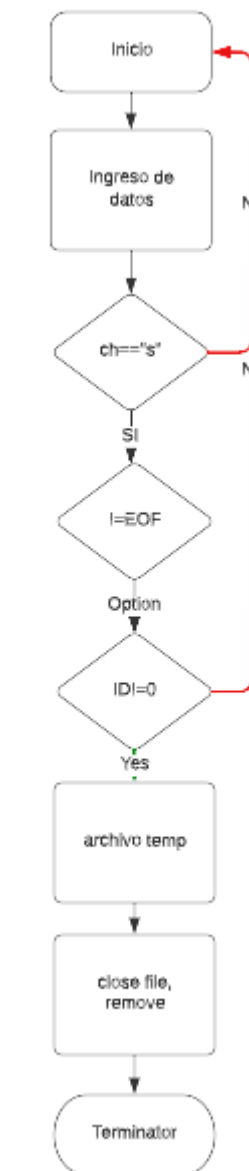
$$V(G) = P + 1$$

$$V(G) = 2 \text{ nodo predicado} + 1 = 3$$

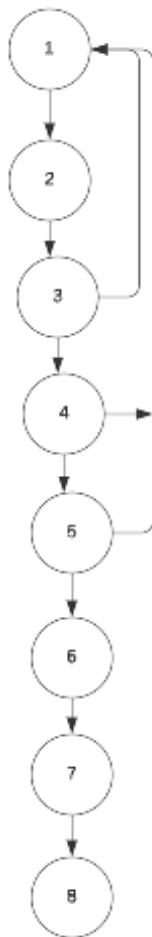
## Prueba Caja Blanca Eliminar Lista de Empleados

```
void deleteEmployeeDetails(void){ //removing records
system("cls");
char checkId[5];
cout<<"\n-----";
cout<<"\nIngrese la identificación del empleado para eliminar: ";
cin>>checkId;
char ch;
cout<<"\n-----";
cout<<"\n\n\nCONFIRMACIÓN\nIngrese 'S' para confirmar la eliminación \n";
cin>>ch;
if(ch=='S'){
    FILE *file, *tempfile;
    file= fopen("data.txt", "r");
    tempfile= fopen("temp.txt", "w");
    while(fscanf(file, "%s %s %s %d %d %d", &name[0], &id[0] , &designation[0], &age, &ctc, &exp) != 0){
        if(strcmp(checkId, id)!=0)
            fprintf(tempfile, "%s %s %s %d %d %d \n", name, id, designation, age, ctc, experienc
        fclose(file);
        fclose(tempfile);
        int isRemoved= remove("data.txt");
        int isRenamed= rename("temp.txt", "data.txt");
        cout<<"\nEliminado con éxito\n";
        waitForEnter();
    }
}
```

### DIAGRAMA DE FLUJO



## GRAFO



### RUTAS:

**R1: 1,2,3,1**

R2: 1,2,3,4,5,1

R3: 1,2,3,4,1

R4: 1,2,3,4,5,6,7,8

### Complejidad Ciclomática

E: Número de aristas

N: Número de nodos

P: Número de nodos predicado

$$V(G) = E - N + 2$$

$$V(G) = 10 - 8 + 2 = 4$$

$$V(G) = P + 1$$

$$V(G) = 3 \text{ nodo predicado} + 1 = 4$$