Logotipo

Descripción generada automáticamente

# Tabla hash

# 

**Fecha:** 14/03/2023

**Materia:** Estructura de datos

**Profesor/Profesora:** Ismael López Buenrostro

**Nombre:** Luis Rigoberto Ramírez Llamas

**Registro:** 21100298

**Carrera:** Drllo. Software

**Grupo:** 4E1

**Descripción del programa**

Este programa es un código en C++ a través de visual studio, en donde se introduce un conjunto de datos en los que se incluyen nombre, apellido, registro y edad sean ordenados y agrupados a través del registro como llave tanto para introducir como para buscar

En este código se usa una clase Tabla Hash y funciones Leer (Para introducir los datos) y Buscar (Para buscar los datos) a través de métodos y struct

**Lista de requerimientos**

Diseñar un programa que implemente una tabla hash, bajo las siguientes características.

a)El programa permitirá la inserción y búsqueda de datos.

b)Se podrán ingresar de 1 –50 valores.

c)Los datos que se ingresarán son estructuras u objetos, con lo siguientes campos o propiedades.

a. Nombre (Cadena de 50 caracteres).

b. Apellido(Cadena de 50 caracteres).

c. Registro (Entero positivo)

d. Edad (Entero positivo)

d)La tabla se creará empleando el registro como llave.

e)La tabla hash contendrá113 elementos.

f)En caso de una colisión los datos se apilarán(pueden estar 2 o más en la misma posición).

g)En el caso de búsqueda: se ingresará un registro y este será buscado en caso de ser encontrado mostrará el nombre, apellido, registro y edad, y en caso de no localizarse se enviará un mensaje de elemento no encontrado.

**Contenido del PSP**

|  |  |  |  |
| --- | --- | --- | --- |
| Código fuente: | Estimado  (min) | Real  (min) | Error |
| Tiempo de Diseño | 120 | 300 | 60% |
| Tiempo de Codificación. | 720 | 1440 | 100% |
| Tiempo estimado de fabricación | 840 | 1740 | 107% |
| Total, de líneas de código nuevas. | 150 | 112 | 25% |
| Total, de líneas de código reusadas. | 0 | 0 | 0% |
| Total, de líneas de código del programa. | 150 | 112 | 4% |
| Cantidad de errores de compilación. | 30 | 20 | 33% |
| Cantidad de errores de ejecución. | 15 | 9 | 40% |

**Código Fuente**

#include <iostream>

#include <string>

#include <stack>

using namespace std;

struct PersonaDatos

{

char Nombre[50];

char Apellido[50];

int Registro;

int Edad;

};

class TablaHash

{

public:

stack<PersonaDatos> Tabla[113];

void Leer(PersonaDatos Datos);

void Buscar(int registro);

};

void TablaHash::Leer(PersonaDatos Datos)

{

Tabla[Datos.Registro % 113].push(Datos);

}

void TablaHash::Buscar(int registro)

{

stack<PersonaDatos> Pos = Tabla[(registro % 113)];

if (!Pos.empty())

{

while (!Pos.empty())

{

PersonaDatos Datos = Pos.top();

if (Datos.Registro == registro)

{

cout << "Nombre: " << Datos.Nombre << endl;

cout << "Apellido: " << Datos.Apellido << endl;

cout << "Registro: " << Datos.Registro << endl;

cout << "\nEdad: " << Datos.Edad << endl;

Pos.pop();

}

}

}

else

{

cout << "\nElemento no encontrado.\n" << endl;

}

};

int main()

{

int Registro = 0;

TablaHash Tabla;

char R = 0;

while (R != 'C')

{

cout << "\nQue quieres hacer?\n[A] Introducir datos\n[B] Buscar\n[C] Salir" << endl;

cin >> R;

switch (R)

{

case 'A':

{

cout << "Se eligio [A]" << endl;

PersonaDatos Datos;

cout << "Ingrese el nombre: " << endl;

fgets(Datos.Nombre, 50, stdin);

fgets(Datos.Nombre, 50, stdin);

cout << "Ingrese el apellido: " << endl;

fgets(Datos.Apellido, 50, stdin);

cout << "Ingrese el registro: " << endl;

cin >> Datos.Registro;

if (Datos.Registro <= 0)

{

cout << "El registro entrado no es valido" << endl;

while (Datos.Registro > 0)

{

cout << "Ingrese el registro: " << endl;

cin >> Datos.Registro;

}

}

cout << "Ingrese la edad: " << endl;

cin >> Datos.Edad;

if (Datos.Edad <= 0)

{

cout << "La edad entrada no es valido" << endl;

while (Datos.Edad > 0)

{

cout << "Ingrese la edad: " << endl;

cin >> Datos.Edad;

}

}

Tabla.Leer(Datos);

break;

}

case 'B':

{

cout << "Se eligio [B]" << endl;

cout << "Ingrese el registro a buscar: ";

cin >> Registro;

Tabla.Buscar(Registro);

break;

}

case 'C':

{

cout << "Se eligio [C]\nSaliendo del programa" << endl;

break;

}

}

if (R > 68 || R < 64)

{

cout << "Opcion no disponible, inrese lo suguerido" << endl;

R = 0;

}

};

return 0;

};