

Actividad 6.3

Dispersión abierta

Algoritmos y Estructuras de Datos

Tema 6: tablas hash

Modifica la actividad 6.2, usando ahora dispersión abierta con listas enlazadas para solucionar las colisiones. Ten en cuenta lo siguiente:

- Eliminaremos el método que nos dice si la tabla está llena, ya que ahora nunca se puede llenar. Igualmente eliminaremos todas las precondiciones relacionadas con este hecho.
- Para practicar, ahora cada estudiante irá en un struct llamado "Estudiante". Dentro del struct pondremos los campos "clave" (tipo int) y "nombre" (tipo string).
- Ahora la tabla hash será un vector de listas enlazadas. Para implementar las listas enlazadas, reutiliza la actividad 4.4 resuelta (lista doblemente enlazada y circular). Tendremos que modificarla un poco:
 - El elemento de cada nodo de la lista enlazada será ahora un struct "Estudiante" en vez de un int.
 - Modifica en consecuencia los métodos necesarios de ListaEnlazada para que ahora los elementos de la lista enlazada sean estudiantes en vez de ints.
 - Modifica también el método "buscar()" de la lista enlazada para que busque elementos en la lista únicamente basándose en la clave.
 - Modifica el método "imprimirListaEnlazada()" para que, al imprimir cada estudiante, imprima entre paréntesis el par (clave,valor). Mira la prueba de caja negra más abajo.
- Plantéate lo siguiente: ¿tienen ahora sentido los flags indicadores de si un elemento de la tabla hash está vacío o ha sido borrado? Si no tienen sentido, elimínalos. En caso contrario, piensa cómo usarlos y dónde ponerlos.
- Nuestra función de hashing calculará dos posiciones e insertará el elemento en la posición que tenga menos elementos enlazados. Una de las posiciones será la función resto, al igual que en los ejercicios anteriores. Piensa cuál sería la otra para que los elementos estén distribuidos lo más uniformemente posible por la tabla. Una posible solución sería la posición simétrica a la primera. Ejemplo: si la tabla tiene 10 posiciones (de 0 a 9) y la primera posición es 8, la simétrica está 5 posiciones más adelante (recuerda que el siguiente del último es el primero), es decir en la posición 3.
- Habrá que actualizar el método "imprimir" para que muestre todo correctamente. Recuerda que este método viola el principio de separación entre modelo e interfaz.
- Al igual que en 6.2, habrá que documentar la complejidad espacial y temporal en el mejor y peor caso.
- Intenta hacer todo de la forma más eficiente posible.

El código del main será similar al del 6.2. La prueba será la siguiente:



```
C:\WINDOWS\system32\cmd.exe
Introduce la capacidad de la tabla: 5
0. Salir
1. Imprimir tabla
2. Comprobar si esta un estudiante
3. Ver un estudiante
4. Meter un estudiante
5. Eliminar un estudiante
Introduzca opcion: 1
Posicion 0: n=0;ListaEnlazada=vacía
Posicion 1: n=0;ListaEnlazada=vacía
Posicion 2: n=0;ListaEnlazada=vacía
Posicion 3: n=0;ListaEnlazada=vacía
Posicion 4: n=0;ListaEnlazada=vacía
0. Salir
1. Imprimir tabla
2. Comprobar si esta un estudiante
3. Ver un estudiante
4. Meter un estudiante
5. Eliminar un estudiante
Introduzca opcion: 4
Introduzca el numero de matricula del estudiante: 101
Introduzca el nombre del estudiante: pepe1
0. Salir
1. Imprimir tabla
2. Comprobar si esta un estudiante
3. Ver un estudiante
4. Meter un estudiante
5. Eliminar un estudiante
Introduzca opcion: 1
Posicion 0: n=0;ListaEnlazada=vacía
Posicion 1: n=1;ListaEnlazada=(101,pepe1),
Posicion 2: n=0;ListaEnlazada=vacía
Posicion 3: n=0;ListaEnlazada=vacía
Posicion 4: n=0;ListaEnlazada=vacía
0. Salir
1. Imprimir tabla
2. Comprobar si esta un estudiante
3. Ver un estudiante
4. Meter un estudiante
5. Eliminar un estudiante
Introduzca opcion: 4
Introduzca el numero de matricula del estudiante: 116
Introduzca el nombre del estudiante: pepe2
0. Salir
1. Imprimir tabla
2. Comprobar si esta un estudiante
3. Ver un estudiante
4. Meter un estudiante
5. Eliminar un estudiante
Introduzca opcion: 1
Posicion 0: n=0;ListaEnlazada=vacía
Posicion 1: n=1;ListaEnlazada=(101,pepe1),
Posicion 2: n=0;ListaEnlazada=vacía
Posicion 3: n=1;ListaEnlazada=(116,pepe2),
Posicion 4: n=0;ListaEnlazada=vacía
0. Salir
1. Imprimir tabla
2. Comprobar si esta un estudiante
3. Ver un estudiante
4. Meter un estudiante
5. Eliminar un estudiante
Introduzca opcion: 4
Introduzca el numero de matricula del estudiante: 103
Introduzca el nombre del estudiante: pepe3
0. Salir
1. Imprimir tabla
2. Comprobar si esta un estudiante
3. Ver un estudiante
4. Meter un estudiante
5. Eliminar un estudiante
Introduzca opcion: 1
Posicion 0: n=1;ListaEnlazada=(103,pepe3),
Posicion 1: n=1;ListaEnlazada=(101,pepe1),
Posicion 2: n=0;ListaEnlazada=vacía
Posicion 3: n=1;ListaEnlazada=(116,pepe2),
Posicion 4: n=0;ListaEnlazada=vacía
0. Salir
1. Imprimir tabla
2. Comprobar si esta un estudiante
3. Ver un estudiante
4. Meter un estudiante
5. Eliminar un estudiante
Introduzca opcion: 4
Introduzca el numero de matricula del estudiante: 218
Introduzca el nombre del estudiante: pepe4
```

```

Introduzca el nombre del estudiante: pepe4
0. Salir
1. Imprimir tabla
2. Comprobar si esta un estudiante
3. Ver un estudiante
4. Meter un estudiante
5. Eliminar un estudiante
Introduzca opcion: 1
Posicion 0: n=1!ListaEnlazada=(103,pepe3),
Posicion 1: n=1!ListaEnlazada=(101,pepe1),
Posicion 2: n=0!ListaEnlazada=vacia
Posicion 3: n=2!ListaEnlazada=(218,pepe4),(116,pepe2),
Posicion 4: n=0!ListaEnlazada=vacia
0. Salir
1. Imprimir tabla
2. Comprobar si esta un estudiante
3. Ver un estudiante
4. Meter un estudiante
5. Eliminar un estudiante
Introduzca opcion: 4
Introduzca el numero de matricula del estudiante: 300
Introduzca el nombre del estudiante: pepe5
0. Salir
1. Imprimir tabla
2. Comprobar si esta un estudiante
3. Ver un estudiante
4. Meter un estudiante
5. Eliminar un estudiante
Introduzca opcion: 1
Posicion 0: n=1!ListaEnlazada=(103,pepe3),
Posicion 1: n=1!ListaEnlazada=(101,pepe1),
Posicion 2: n=1!ListaEnlazada=(300,pepe5),
Posicion 3: n=2!ListaEnlazada=(218,pepe4),(116,pepe2),
Posicion 4: n=0!ListaEnlazada=vacia
0. Salir
1. Imprimir tabla
2. Comprobar si esta un estudiante
3. Ver un estudiante
4. Meter un estudiante
5. Eliminar un estudiante
Introduzca opcion: 4
Introduzca el numero de matricula del estudiante: 400
Introduzca el nombre del estudiante: pepe6
0. Salir
1. Imprimir tabla
2. Comprobar si esta un estudiante
3. Ver un estudiante
4. Meter un estudiante
5. Eliminar un estudiante
Introduzca opcion: 1
Posicion 0: n=2!ListaEnlazada=(400,pepe6),(103,pepe3),
Posicion 1: n=1!ListaEnlazada=(101,pepe1),
Posicion 2: n=1!ListaEnlazada=(300,pepe5),
Posicion 3: n=2!ListaEnlazada=(218,pepe4),(116,pepe2),
Posicion 4: n=0!ListaEnlazada=vacia
0. Salir
1. Imprimir tabla
2. Comprobar si esta un estudiante
3. Ver un estudiante
4. Meter un estudiante
5. Eliminar un estudiante
Introduzca opcion: 3
Introduzca el numero de matricula del estudiante: 116
El estudiante se llama pepe2
0. Salir
1. Imprimir tabla
2. Comprobar si esta un estudiante
3. Ver un estudiante
4. Meter un estudiante
5. Eliminar un estudiante
Introduzca opcion: 5
Introduzca el numero de matricula del estudiante: 116
0. Salir
1. Imprimir tabla
2. Comprobar si esta un estudiante
3. Ver un estudiante
4. Meter un estudiante
5. Eliminar un estudiante
Introduzca opcion: 1
Posicion 0: n=2!ListaEnlazada=(400,pepe6),(103,pepe3),
Posicion 1: n=1!ListaEnlazada=(101,pepe1),
Posicion 2: n=1!ListaEnlazada=(300,pepe5),
Posicion 3: n=1!ListaEnlazada=(218,pepe4),
Posicion 4: n=0!ListaEnlazada=vacia

```

```

0. Salir
1. Imprimir tabla
2. Comprobar si esta un estudiante
3. Ver un estudiante
4. Meter un estudiante
5. Eliminar un estudiante
Introduzca opcion: 5
Introduzca el numero de matricula del estudiante: 218
0. Salir
1. Imprimir tabla
2. Comprobar si esta un estudiante
3. Ver un estudiante
4. Meter un estudiante
5. Eliminar un estudiante
Introduzca opcion: 1
Posicion 0: n=2;ListaEnlazada=(400,pepe6),(103,pepe3),
Posicion 1: n=1;ListaEnlazada=(101,pepe1),
Posicion 2: n=1;ListaEnlazada=(300,pepe5),
Posicion 3: n=0;ListaEnlazada=vacia
Posicion 4: n=0;ListaEnlazada=vacia
0. Salir
1. Imprimir tabla
2. Comprobar si esta un estudiante
3. Ver un estudiante
4. Meter un estudiante
5. Eliminar un estudiante
Introduzca opcion: 4
Introduzca el numero de matricula del estudiante: 601
Introduzca el nombre del estudiante: pepe8
0. Salir
1. Imprimir tabla
2. Comprobar si esta un estudiante
3. Ver un estudiante
4. Meter un estudiante
5. Eliminar un estudiante
Introduzca opcion: 1
Posicion 0: n=2;ListaEnlazada=(400,pepe6),(103,pepe3),
Posicion 1: n=1;ListaEnlazada=(101,pepe1),
Posicion 2: n=1;ListaEnlazada=(300,pepe5),
Posicion 3: n=1;ListaEnlazada=(601,pepe8),
Posicion 4: n=0;ListaEnlazada=vacia
0. Salir
1. Imprimir tabla
2. Comprobar si esta un estudiante
3. Ver un estudiante
4. Meter un estudiante
5. Eliminar un estudiante
Introduzca opcion: 0
Presione una tecla para continuar . . .

```