

Archivos Hash con resolución de colisiones por lista en archivo desborde
Andrés Arcia

Crear

1. TAM_MAX_P= cte1*t
2. TAM_MAX_D= cte2*t
3. TAM_ACT_P=0, TAM_ACT_D=0
4. R.hab=verdadero
5. r.sig=-1
6. Abrir PRINCIPAL
7. RP(i=0; i< TAM_MAX_P; i++)
 - 7.1 Escribir (P,i,R)
8. Abrir DESBORDE
9. RP(i=0;i< TAM_MAX_D;i++)
 - 9.1 PILAS_DESBORDE.push(i)
 - 9.2 ESCRIBIR (D,I,R)
10. Cerrar PRINCIPAL, DESBORDE;

Abrir Archivo

1. TAM_MAX_P= TAM_MAX_D= TAM_ACT_P= TAM_ACT_D=0
2. Abrir PRINCIPAL
3. continuar=verdadero
4. RM (continuar=verdadero)
 - 4.1 Si (leer(P,TAM_MAX_P,R)==verdadero)
 - 4.1.1 TAM_MAX_P ++
 - 4.1.2 SI (R.hab==falso)
 - 4.1.2.1 TAM_ACT_P++
 - Sino
 - 4.1.3 continuar=falso
5. Cerrar PRINCIPAL
6. Abrir DESBORDE
7. continuar=verdadero
8. RM (continuar=verdadero)
 - 8.1 Si (leer(D, TAM_MAX_D, R) == verdadero)
 - 8.1.1 TAM_MAX_D ++
 - 8.1.2 SI (R.hab==verdadero)
 - 8.1.2.1 PILA_DESBORDE.push(TAM_MAX_D -1)
 - Sino
 - 8.1.2.2 TAM_ACT_D ++
 - Sino
 - 8.1.3 continuar=falso
9. cerrar DESBORDE

Buscar Registro

R = registro que contiene la clave a buscar

S= registro existente en una posición cualquiera

1. POS= HASH(R.clave)
2. Si (POS<0 v POS> TAM_MAX_P)
 - 2.1. Retornar ERROR
3. Leer (P,POS,S)
4. Si (S.hab==falso)
 - 4.1. Si (R.clave==S.clave)
 - 4.1.1.Cerrar PRINCIPAL
 - 4.1.2.Devolver S
 - Sino
 - 4.1.3.SI (S.sig j=-1)
 - [4.1.3.1.](#) Abrir DESBORDE
 - [4.1.3.2.](#) continuar = verdadero
 - [4.1.3.3.](#) RM (continuar)
 - 4.1.3.3.1. Leer(D,S.sig,S)
 - 4.1.3.3.2. SI (S.sig==1)
 - 4.1.3.3.2.1.continuar=falso
 - 4.1.3.3.3. Si (S.clave==R.clave)
 - 4.1.3.3.3.1.Cerrar PRINCIPAL, DESBORDE
 - 4.1.3.3.3.2.Retornar S
 - [4.1.3.4.](#) Cerrar DESBORDE
5. Cerrar PRINCIPAL

Agregar Registro

1. POS = HASH (R.clave)
2. Si (POS<0 v POS>= TAM_MAX_P)
 - 2.1. Retornar ERROR
3. Preparar el registro (R.data, R.hab=falso, R.sig=-1)
4. Leer (P,POS,S)
5. Si (S.hab==verdadero)
 - 5.1. Escribir (P,POS,R)
 - 5.2. Cerrar PRINCIPAL
 - 5.3. TAM_ACT_P ++
 - 5.4. Retornar ÉXITO
- Sino
- 5.5. Abrir DESBORDE

- 5.6. Si (S.sig== -1)
 - 5.6.1.S.sig=PILA_DESBORDE.pop()
 - 5.6.2.Si (S.sig== -1)
 - [5.6.2.1.](#) Cerrar PRINCIPAL, DESBORDE
 - [5.6.2.2.](#) Retornar ERROR
 - 5.6.3.Escribir (P,POS,S)
 - 5.6.4.Cerrar PRINCIPAL
 - Sino
 - 5.6.5.RM (S.sig j= -1)
 - [5.6.5.1.](#) anterior=S.sig
 - [5.6.5.2.](#) Leer (D,S.sig,S)
 - 5.6.6.S.sig=PILA_DESBORDE.pop()
 - 5.6.7.Si (S.sig== -1)
 - [5.6.7.1.](#) Cerrar DESBORDE
 - [5.6.7.2.](#) Retornar ERROR
 - 5.6.8.Escribir(D,anterior,S)
6. TAM_ACT_D ++
7. Escribir (D,S.sig,R)
8. Cerrar D

Eliminacion

1. pos = hash(k)
2. Si (pos<0 v pos > tam_max_p)
 - 2.1. Retornar ERROR
3. LEER (maestro, pos, &S) // Leer del archivo **maestro** en la posicion pos y depositar en **S**.
4. Si (S.disp == verdadero)
 - 4.1. Cerrar(maestro)
 - 4.2. Retornar ERROR (clave inexistente)
5. Si (clave == S.clave)
 - 5.1. Si (S.prox = -1)
 - 5.1.1. S.disp = verdadero
 - 5.1.2. ESCRIBIR (maestro, pos, S)
 - 5.1.3. cerrar(maestro)
 - 5.1.4. tam_act_maestro --
 - 5.1.5. Retornar (EXITO)
 - SI NO
 - 5.1.6Abrir (desborde)
 - 5.1.7 LEER(desborde, S.prox, T)
 - 5.1.8 ESCRIBIR(principal, pos, T)

- 5.1.9 T.disp = verdadero
- 5.1.10 T.prox = -1
- 5.1.11 ESCRIBIR(desborde, S.prox, T)
- 5.1.12 tam_act_desb --
- 5.1.13 pila_desborde.push(S.prox)
- 5.1.14 Cerrar(maestro, desborde)
- 5.1.15 Retornar (EXITO)
- SI NO (a lo mejor esta en el desborde)
- 5.2 Abrir (Desborde)
- 5.3 Actualizar_maestro = verdadero
- 5.4 Encontrado = falso
- 5.5 Anterior = pos
- 5.6 Actual = S.prox
- 5.7 continuar = verdadero
- 5.8 R.M. (continuar = verdadero ^ encontrado = falso)
 - 5.8.1. Anterior = actual
 - 5.8.2. Actual = S.prox
 - 5.8.3. LEER(desborde, actual, S)
 - 5.8.4. Si (S.prox = -1)
 - 5.8.4.1. continuar = falso
 - 5.8.5 Si (S.clave = clave)
 - 5.8.5.1. encontrado = verdadero
 - SI NO
 - 5.8.5.2 actualizar_ppal = falso
- 5.9 Si (encontrado = falso)
 - 5.9.1. Cerrar (maestro, desborde)
 - 5.9.2. Retornar ERROR
- 5.10 Si (actualizar_principal == verdadero)
 - 5.10.1. LEER (maestro, pos, T)
 - 5.10.2. T.prox = S.prox
 - 5.10.3. ESCRIBIR(maestro, pos, T)
 - SI NO
 - 5.10.4 LEER(desborde, anterior, T)
 - 5.10.5 T.prox = S.prox
 - 5.10.6 ESCRIBIR(desborde, anterior, T)
- 5.11 S.disp = verdadero
- 5.12 S.prox = -1
- 5.13 ESCRIBIR(desborde, actual, S)
- 5.14 tam_act_desb --
- 5.15 pila_desborde.push(actual)
- 5.16 Cerrar (desborde)
- 6. Cerrar (principal)
- 7. Retornar EXITO