DS Lab Assignment 12

Pranav Joshi
CS-B Batch 2
Roll no: 43

Title: WAP to implement Hashing using Linear Probing with chaining without replacement. (Hash Function: Key % table size)

Code:

```c
#include <stdio.h>
#include <conio.h>
#define SIZE 10
#define FALSE 0
#define TRUE 1
#define h(x) x % SIZE
void insert(int data[], int flag[], int chain[], int x);
int search(int data[], int flag[], int chain[], int x);
void print(int data[], int flag[], int chain[]);

void main(){
    int data[SIZE], flag[SIZE], chain[SIZE], i, j, x, loc, num_inputs;
    for (i = 0; i < SIZE; i++){
        flag[i] = FALSE;
        chain[i] = -1;
    }

    printf("\nEnter the number of inputs (up to %d): ", SIZE);
    scanf("%d", &num_inputs);

    printf("\nEnter %d numbers to be inserted:", num_inputs);
    for (i = 0; i < num_inputs; i++){
        scanf("%d", &x);
        insert(data, flag, chain, x);
    }

    printf("\nHash Table Contents:");
    print(data, flag, chain);

    printf("\n\nSearch Results:\n");
    for (i = 0; i < num_inputs; i++){
```

```
        printf("Enter a number to be searched: ");
        scanf("%d", &x);
        loc = search(data, flag, chain, x);
        if (loc == -1)
            printf("***Element %d not found***\n", x);
        else
            printf("***Found %d at location %d***\n", x, loc);
    }
}


void insert(int data[], int flag[], int chain[], int x){
    int i = 0, j, start;
    start = h(x);
    while (flag[start] && i < SIZE){
        if (data[start] % SIZE == x % SIZE)
            break;
        i++;
        start = (start + 1) % SIZE;
    }
    if (i == SIZE){
        printf("\n**hash table is full***");
        return;
    }
    while (chain[start] != -1)
        start = chain[start];
    j = start;
    while (flag[j] && i < SIZE){
        j = (j + 1) % SIZE;
        i = i + 1;
    }
    if (i == SIZE)
    {
        printf("\n**hash table is full***");
        return;
    }
    data[j] = x;
    flag[j] = TRUE;
    if (j != start)
        chain[start] = j;
}

int search(int data[], int flag[], int chain[], int x){
    int i = 0, j;
    j = h(x);
    while (i < SIZE && flag[j] && data[j] % SIZE != x % SIZE){
        i++;
        j = (j + 1) % SIZE;
    }
```

```
    if (!flag[j] || i == SIZE)
        return (-1);
    while (j != -1)
    {
        if (data[j] == x)
            return (j);
        j = chain[j];
    }
    return (-1);
}

void print(int data[], int flag[], int chain[]){
    int i;
    for (i = 0; i < SIZE; i++)
        if (flag[i])
            printf("\n(%d) %d     %d", i, data[i], chain[i]);
        else
            printf("\n(%d) ---    %d", i, chain[i]);
}
```

Output:

```
C:\Code\C\Code\Sem4Assignment12.exe

Enter the number of inputs (up to 10): 8

Enter 8 numbers to be inserted:15 5 28 8 19 25 16 17

Hash Table Contents:
(0) 19     -1
(1) 16     -1
(2) 17     -1
(3) ---    -1
(4) ---    -1
(5) 15     6
(6) 5     7
(7) 25     -1
(8) 28     9
(9) 8     -1

Search Results:
Enter a number to be searched: 15
***Found 15 at location 5***
Enter a number to be searched: 9
***Element 9 not found***
```