1BM22CS219 – RANJAN DEVI

WEEK 1

1 Develop a C program that simulates a basic banking system with functionalities like account creation, withdrawal, deposit, and balance inquiry. Write different user-defined function for each.

#include <stdio.h>

#include <stdlib.h>

void ac\_create(double);

void withdrawal(double);

void deposit(double);

void balance(double);

void main()

{

double b=0;

printf("Let's create your account\n");

ac\_create(b);

}

void ac\_create(double b)

{

char name [10];

int num;

int flag;

printf("Enter your name: ");

scanf("%s", name);

printf("Enter your phone number:");

scanf("%d", &num);

printf("Account created successfully!\n");

deposit(b);

}

void deposit (double b)

{

double d;

int flag;

printf("Enter the amount to be deposited\n");

scanf("%lf",&d);

b=b+d;

printf("The amount is deposited succesfully\n");

printf("Press 1 to withdraw some money\n");

scanf("%d", &flag);

if(flag==1)

withdrawal (b);

else

{

printf("Your balance is:%lf",b);

exit(0);

}

}

void withdrawal(double b)

{

double w;

int flag;

printf("Enter the amount to be withdrawed: \n");

scanf("%lf",&w);

b=b-w;

printf("Press 1 to know the balance\n");

scanf("%d", &flag);

if(flag==1)

balance(b);

else

exit(0);

}

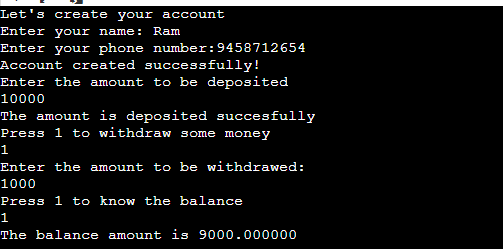
void balance(double b)

{

printf("The balance amount is %lf",b);

}

OUTPUT:



2. Implement a C program that sorts strings lexicographically, considering uppercase and lowercase letters, and without using the standard library sorting functions.

#include <stdio.h>

#include <stdlib.h>

Void main()

{

char str[5][50], temp[50];

printf("Enter 5 words: ");

for (int i = 0; i < 5; ++i)

fgets(str[i], sizeof(str[i]), stdin);

for (int i = 0; i < 5; ++i)

{

for (int j = i + 1; j < 5; ++j)

{

if (strcmp(str[i], str[j]) > 0)

{ strcpy(temp, str[i]);

strcpy(str[i], str[j]);

strcpy(str[j], temp);

}

}

}

printf("In the lexicographical order: \n");

for (int i = 0; i < 5; ++i)

fputs(str[i], stdout);

}

OUTPUT:

Enter 5 words: R programming

JavaScript

Java

C programming

C++ programming

In the lexicographical order:

C programming

C++ programming

Java

JavaScript

R programming

3. Implement a C program to check if a given element is present in a 2D array with a user defined function.

#include <stdio.h>

#include <stdlib.h>

void search(int \*,int \*,int \*,int \*);

void main()

{

int a[10][10];

int i,j,n,m,key;

printf("Enter order of array:");

scanf("%d%d",&m,&n);

printf("Enter elements:\n");

for(i=0;i<m;i++)

{

for(j=0;j<n;j++)

scanf("%d",&a[i][j]);

}

printf("Enter key:");

scanf("%d",&key);

search(a,&key,&n,&m);

}

void search(int \*array, int \*key, int \*n, int\*m)

{

int i,j;

for(i=0;i<\*m;i++)

{

for(j=0;j<\*n;j++)

{

if(\*key==array[i])

{ printf("Key is present");

exit(0);

}

}

}

printf("Key is not present");

}

OUTPUT:

Enter order of array: 2 2

Enter elements:

2 4 8 10

Enter key: 4

Key is present

5. Write a C program to find the index of the last occurrence of a number in an array with a user defined function.

#include <stdio.h>

#include <stdlib.h>

void occurence(int \*,int \*,int \*);

void main()

{

int a[10];

int i,j,n,m,key;

printf("Enter order of array:");

scanf("%d",&n);

printf("Enter elements:\n");

for(i=0;i<n;i++)

{

scanf("%d",&a[i]);

}

printf("Enter key:");

scanf("%d",&key);

occurence(a,&key,&n);

}

void occurence(int \*array, int \*key, int \*n)

{

int i,last=-1;

for(i=0;i<\*n;i++)

{

if(\*key!=array[i])

continue;

last=i;

}

printf("Last occurence= %d",last);

}

OUTPUT:

Enter no of elements: 8

Enter elements:

2 4 6 4 7 3 4 4

Enter key: 4

Last occurrence= 7

6. Write a program to search for a specific element in an array using linear search.

#include <stdio.h>

#include <stdlib.h>

void search(int \*,int \*,int \*);

void main()

{

int a[10];

int i,n,key;

printf("Enter no of elements:");

scanf("%d",&n);

printf("Enter elements:\n");

for(i=0;i<n;i++)

{

scanf("%d",&a[i]);

}

printf("Enter key:");

scanf("%d",&key);

search(a,&key,&n);

}

void search(int \*array, int \*key, int \*n)

{

int i;

for(i=0;i<\*n;i++)

{

if(\*key==array[i])

{ printf("Key found at %d",i);

exit(0);

}

}

printf("Key not found");

}

OUTPUT-

Enter no of elements: 5

Enter elements:

2 4 6 8 10

Enter key: 6

Key found at 3 position

7. Write a program to perform binary search on sorted array with a user defined function.

#include <stdio.h>

#include <stdlib.h>

void search(int \*,int \*,int \*);

void main()

{

int a[10];

int i,n,key;

printf("Enter no of elements:");

scanf("%d",&n);

printf("Enter elements:\n");

for(i=0;i<n;i++)

{

scanf("%d",&a[i]);

}

printf("Enter key:");

scanf("%d",&key);

search(a,&key,&n);

}

void search(int \*ar, int \*key, int \*n)

{

int mid;

int low=0,high=\*n-1;

while(low<=high)

{

mid=(low+high)/2;

if(\*key==ar[mid])

{

printf("Key found at %d position",mid+1);

exit(0);

}

if(\*key>ar[mid])

low=mid+1;

if(\*key<ar[mid])

high=mid-1;

}

printf("Key not found");

}

OUTPUT:

Enter no of elements: 5

Enter elements:

1 3 4 5 8

Enter key: 5

Key found at 4 position

8. Create a program in C to search for the minimum and maximum elements in an array with a user defined function.

#include <stdio.h>

#include <stdlib.h>

void min\_max(int \*,int \*);

void main()

{

int a[10];

int i,n,key;

printf("Enter no of elements:");

scanf("%d",&n);

printf("Enter elements:\n");

for(i=0;i<n;i++)

{

scanf("%d",&a[i]);

}

min\_max(a,&n);

}

void min\_max(int \*arr, int \*n)

{

int max = arr[0];

int min = arr[0];

int i;

for(i=0; i<\*n; i++)

{

if(arr[i] > max)

{

max = arr[i];

}

if(arr[i] < min)

{

min = arr[i];

}

}

printf("Maximum element = %d\n", max);

printf("Minimum element = %d",min);

}

OUTPUT:

Enter no of elements: 5

Enter elements:

2 7 1 5 8

Maximun element = 8

Minimum element = 1