**VIVEKANANDA INSTITUTE OF PROFESSIONAL STUDIES**

**VIVEKANANDA SCHOOL OF INFORMATION TECHNOLOGY**



**BACHELOR OF COMPUTER APPLICATION**

**PROGRAMMING USING ‘C’ LANGUAGE LAB FILE**

**BCA 171**

**Guru Gobind Singh Indraprastha University   
Sector - 16C, Dwarka, Delhi - 110078**



**SUBMITTED TO: SUBMITTED BY:**

Ms. Swati Jain Karthik Nair

Assistant Professor

VSIT

**INDEX**

|  |  |  |
| --- | --- | --- |
| **S. No.** | **Program** | **Page no.** |
| **1** | **Write a program to convert temperature from Celsius to Fahrenheit by taking input from the user.** | **5** |
| **2** | **Write a program to find the greatest number among 3 numbers given by the user** | **6** |
| **3** | **Write a program to check if a given number is prime or not.** | **8** |
| **4** | **Write a program to display the following pattern up to N rows, taking the value of N from the user:**  **1**  **2 3**  **4 5 6**  **7 8 9 10** | **10** |
| **5** | **write a program to input marks of 50 students using an array and display the array in ascending order.** | **11** |
| **6** | **Write a program to search for a number entered by the user in a given array and display the array in ascending order.** | **13** |
| **7** | **Write a program to check if a string in palindrome or not.** | **15** |
| **8** | **Write a program to add, subtract, multiply and divide two numbers using pointers.** | **17** |
| **9** | **Write a program to create a structure for employees containing the following data members: Employee ID, Employee Name, Age, Address, Department and Salary.**  **Input data for 10 employees and display the details of the employee from the employee ID given by the user** | **18** |
| **10** | **Write a program to create two files with names EvenFile and OddFile. Input 20 numbers from the user and save even numbers in EvenFile and odd numbers in OddFile.** | **21** |
| **11** | **Write a menu driven program to construct a calculator for following arithmetic operations: addition, subtraction, multiplication,**  **Division, average.** | **22** |
| **12** | **Write a menu driven program to perform the following operations:**  **(i) Print armstrong numbers upto N**  **(ii) Display prime numbers between 1 to N**  **(iii) Reverse of an integer** | **27** |
| **13** | **Write a program to convert hexadecimal number into a binary number.** | **31** |
| **14** | **Write a program to calculate factorial of a number and display Fibonacci series up to N terms using recursive functions** | **34** |
| **15** | **Write a program to perform matrix addition, matrix multiplication, and matrix transpose on 2-d arrays.** | **36** |

**Q1. Write a program to convert Celcius to Farenheit by taking input from user.**

**Solution:**

// Convert celsius to fahrenheit by taking input from the user

#include <stdio.h>

int main()

{

int celsius;

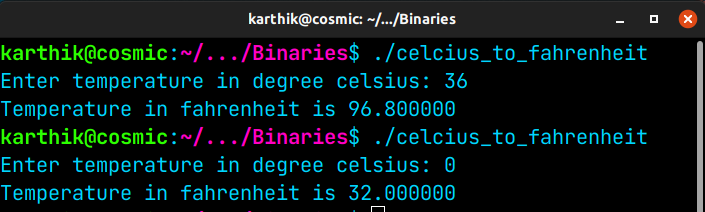
printf("Enter temperature in degree celsius: ");

scanf("%d", &celsius);

printf("Temperature in fahrenheit is %f\n", (celsius\*1.8)+(32));

}

**Output:**



**Q2. Write a program to find the greatest numbers among 3 numbers given by user**

**Solution:**

// Write a C program to find the greatest numbers among 3 numbers given by user

#include <stdio.h>

int main()

{

int a,b,c;

printf("Enter the first number: ");

scanf("%d", &a);

printf("Enter the second number: ");

scanf("%d", &b);

printf("Enter the third number: ");

scanf("%d", &c);

if (((a>b)&&(a>c))||((a==b)&&(a>c)))

{

printf("%d is the largest number !",a);

}

else if (((b>a)&&(b>c))||((b==c)&&(b>a)))

{

printf("%d is the largest number !",b);

}

else if (((c>a)&&(c>b))||((a==c)&&(a>b)))

{

printf("%d is the largest number !",c);

}

else

{

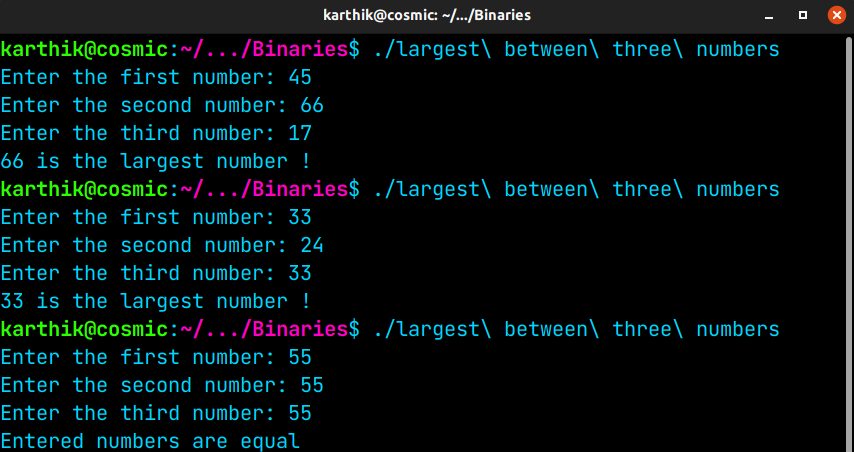
printf("Entered numbers are equal");

}

printf("\n");

}

**Output:**



**Q3. Write a program check if a number is prime or not**

**Solution:**

// Find if entered number is a prime number

#include <stdio.h>

int prime\_num(int number);

int prime\_num(int number)

{

int i, j=(number/2);

for (i=2;i<=j;i++)

{

if (number%i==0)

{

return 0;

}

}

return 1;

}

int main()

{

int n; int if\_prime;

printf("Enter a number: ");

scanf("%d", &n);

if (n<0)

{

printf("Negative numbers can't be prime!\n"); return 0;

}

else if (n<=1)

{

printf("%d is neither prime nor composite!\n", n); return 0;

}

else

{

if\_prime=prime\_num(n);

if (if\_prime==1)

{

printf("%d is a prime number\n", n);

}

else

{

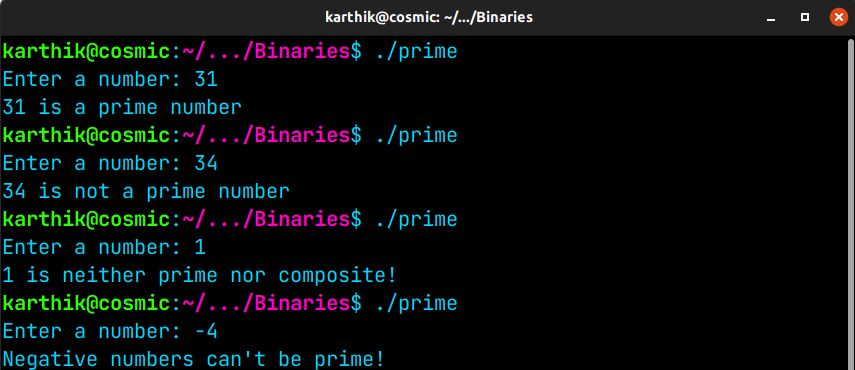
printf("%d is not a prime number\n", n);

}

}

}

**Output:**



**Q4. Write a program to display the following pattern upto n**

**1**

**2 3**

**4 5 6**

**7 8 9 10**

**Solution:**

#include <stdio.h>

int main()

{

int i=1, j=1, n=1, r;

printf("Enter value of n: ");

scanf("%d", &r);

for (i=1; i<=r; i++)

{

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

{

printf("%d ", n);

n++;

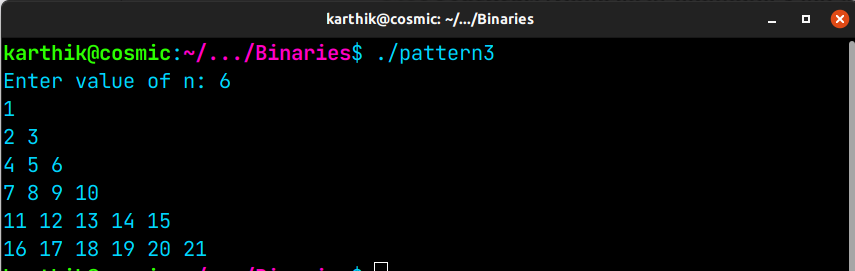
}

printf("\n");

}

}

**Output:**



**Q5. Write a program to input marks of 50 students using an array and display the average marks of the class**

**Solution:**

// Write a program to input marks of 50 students using an array and display the average marks of the class.

#include <stdio.h>

int main()

{

float sum=0, average=0;

int students[50], i, j;

for (i = 0; i < 50; i++)

{

printf("Input marks of student %d: ", i+1);

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

}

for (i = 0; i < 50; i++)

{

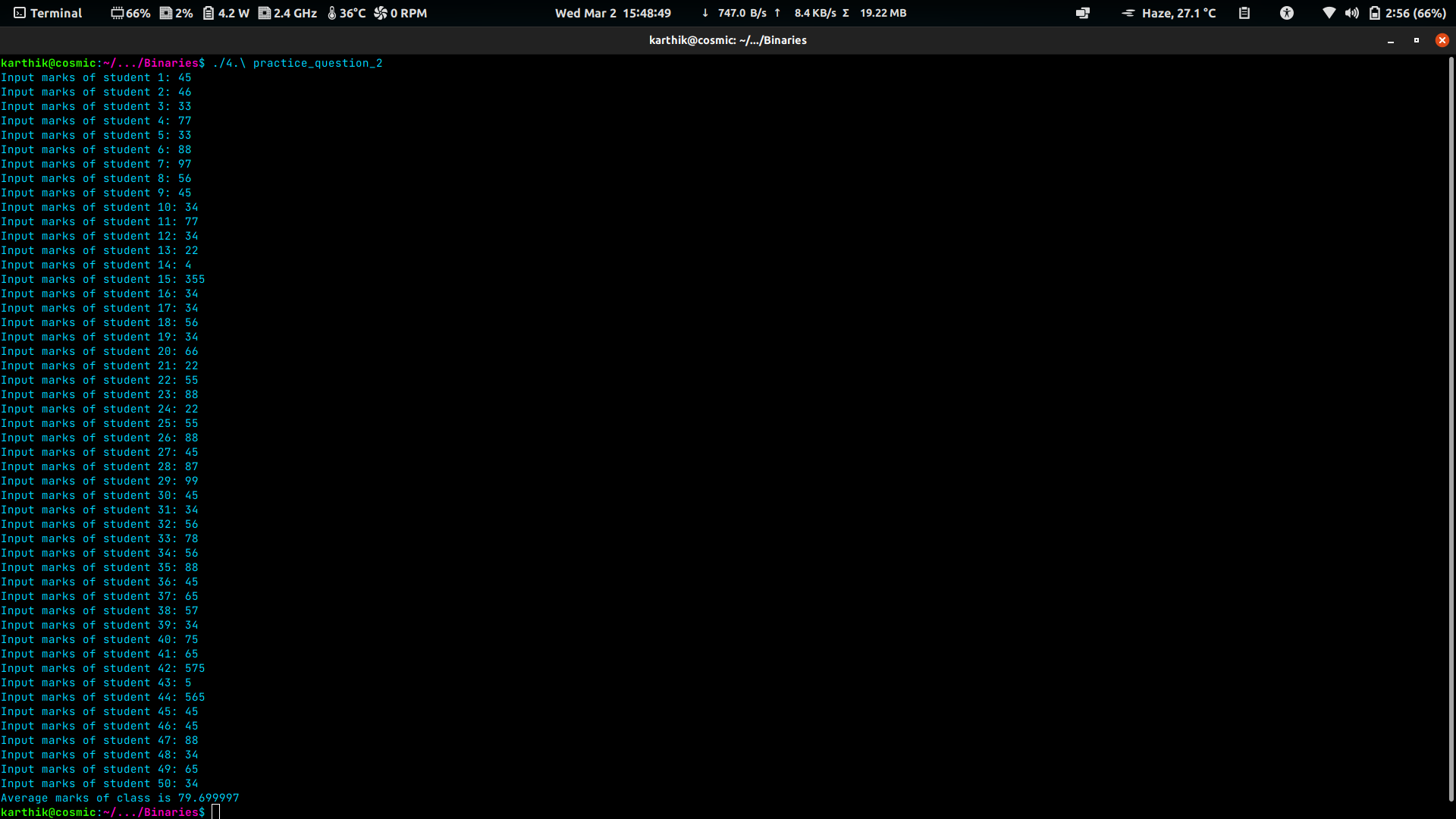
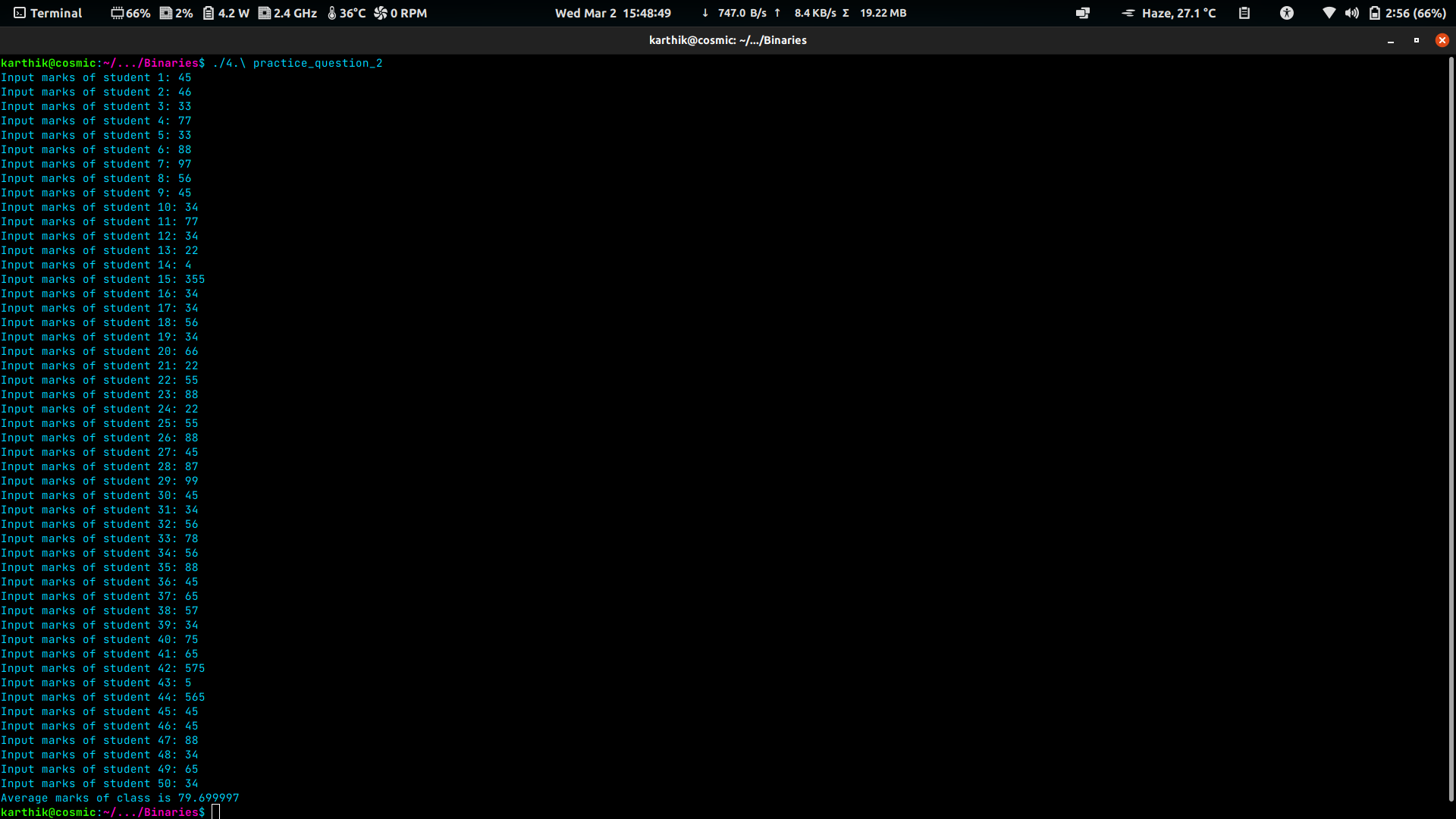
sum += students[i];

}

printf("Average marks of class is %f\n", sum/50);

}

**Output:**

****

**Q6. Write a program to search for a number entered by the user in a given array and display the array in ascending order.**

**Solution:**

// Write a program to search for a number entered by the user in a given array and display the array in ascending order

#include <stdio.h>

int main()

{

int array[]={3,4,43,23,32,33,21,39,98};

int i,j,n,flag=0, temp;

printf("Enter the number to be searched: ");

scanf("%d", &n);

for (i=0; i<9; i++)

{

if (n==array[i])

{

printf("%d found in array at index %d\n", n,i);

flag=1;

}

}

if(flag==0)

{

printf("%d not found in array!\n",n);

}

// Sorting the array

for (i=0; i<9; i++)

{

for (j=i+1;j<9;j++)

{

if (array[i]>array[j])

{

temp=array[i];

array[i]=array[j];

array[j]=temp;

}

}

}

// Displaying the array in ascending order

printf("The array in ascending order is\n");

for (i=0; i<9; i++)

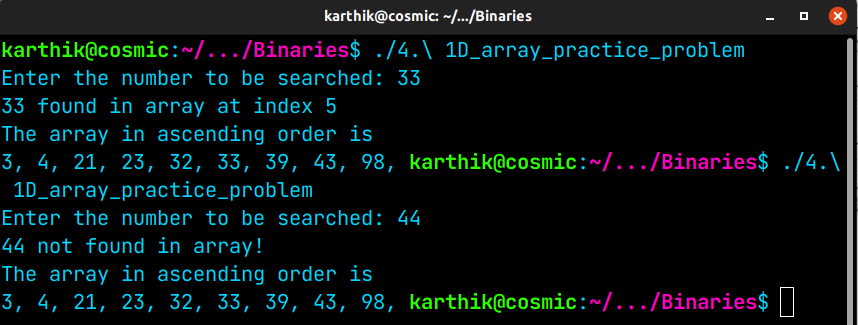
{

printf("%d, ", array[i]);

}

}

**Output:**

****

**Q7. Write a program to check if a string is palindrome or not**

**Solution:**

// Write a program to check if a string is palindrome or not

#include <stdio.h>

#include <string.h>

int main()

{

int i=0, j=0, flag=1;

char text[100];

printf("Enter string: ");

fgets(text, 100, stdin);

int len=strlen(text);

for (i=(len-2); i>=0; i--)

{

if(text[i]!=text[j])

{

flag=0; i=-1;

}j++;

}

if(flag==1)

{

printf("Text is palindrome\n");

}

else

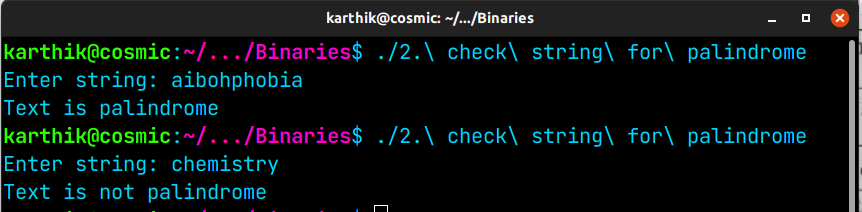
{

printf("Text is not palindrome\n");

}

}

**Output:**



**Q8. Write a program to add, subtract, multiply and divide two numbers using pointers**

**Solution:**

**// Write a program to add, subtract, multiply and divide two numbers using pointers**

#include <stdio.h>

int main()

{

int a, b;

printf("Enter first number: ");

scanf("%d", &a);

printf("Enter second number: ");

scanf("%d", &b);

int\* ptr1= &a;

int\* ptr2= &b;

printf("%d added to %d is %d\n", \*ptr1, \*ptr2, \*ptr1+\*ptr2);

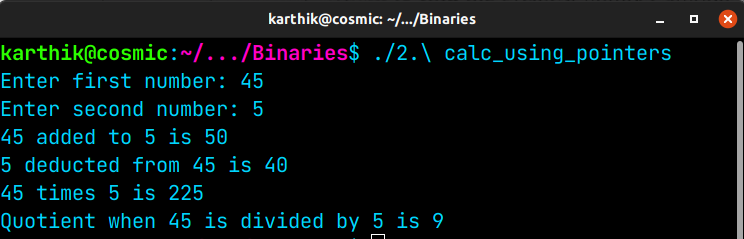
printf("%d deducted from %d is %d\n", \*ptr2, \*ptr1, \*ptr1-\*ptr2);

printf("%d times %d is %d\n", \*ptr1, \*ptr2, \*ptr1\*\*ptr2);

printf("Quotient when %d is divided by %d is %d\n", \*ptr1, \*ptr2, (\*ptr1)/(\*ptr2));

}

**Output:**



**Q9. Write a program to create a structure for employees containing the following data members: Employee ID, Employee Name, Age, Address, Department and Salary.**

**Input data for 10 employees and display the details of the employee from the employee ID given by the user.**

**Solution**

#include <stdio.h>

#include <string.h>

void discard\_remainder\_of\_line( void )

{

int c;

// discarding the leftovers of a scanf

do

{

c = getchar();

} while ( c != EOF && c != '\n' );

}

int main()

{

int i=0;

typedef struct employee

{

long int id;

char name[30];

int age;

char address[100];

char dept[30];

long int salary;

}emp;

emp list[5];

for (i = 0; i<10;i++)

{

printf("Enter id of employee %d : ", i+1);

scanf("%ld", &(list[i].id));

discard\_remainder\_of\_line();

printf("Enter name of employee %d : ", i+1);

fgets(list[i].name, sizeof(list[i].name), stdin);

printf("Enter age of employee %d : ", i+1);

scanf("%d", &(list[i].age));

discard\_remainder\_of\_line();

printf("Enter address of employee %d : ", i+1);

fgets(list[i].address, sizeof(list[i].address), stdin);

printf("Enter dept of employee %d : ", i+1);

fgets(list[i].dept, sizeof(list[i].dept), stdin);

printf("Enter salary of employee %d : ", i+1);

scanf("%ld", &list[i].salary);

discard\_remainder\_of\_line();

}

long int emp\_id;

printf("\nEnter employee id to be searched: ");

scanf("%ld", &emp\_id);

discard\_remainder\_of\_line();

// Linear search

for (i=0; i<10; i++)

{

if (list[i].id==emp\_id)

{

printf("\nEmployee found!!\n");

printf("Id : %ld \n", list[i].id);

printf("Name : %s \n", list[i].name);

printf("Age : %d\n", list[i].age);

printf("Address : %s \n", list[i].address);

printf("Dept : %s \n", list[i].dept);

printf("Salary : %ld \n", list[i].salary);

return 0;

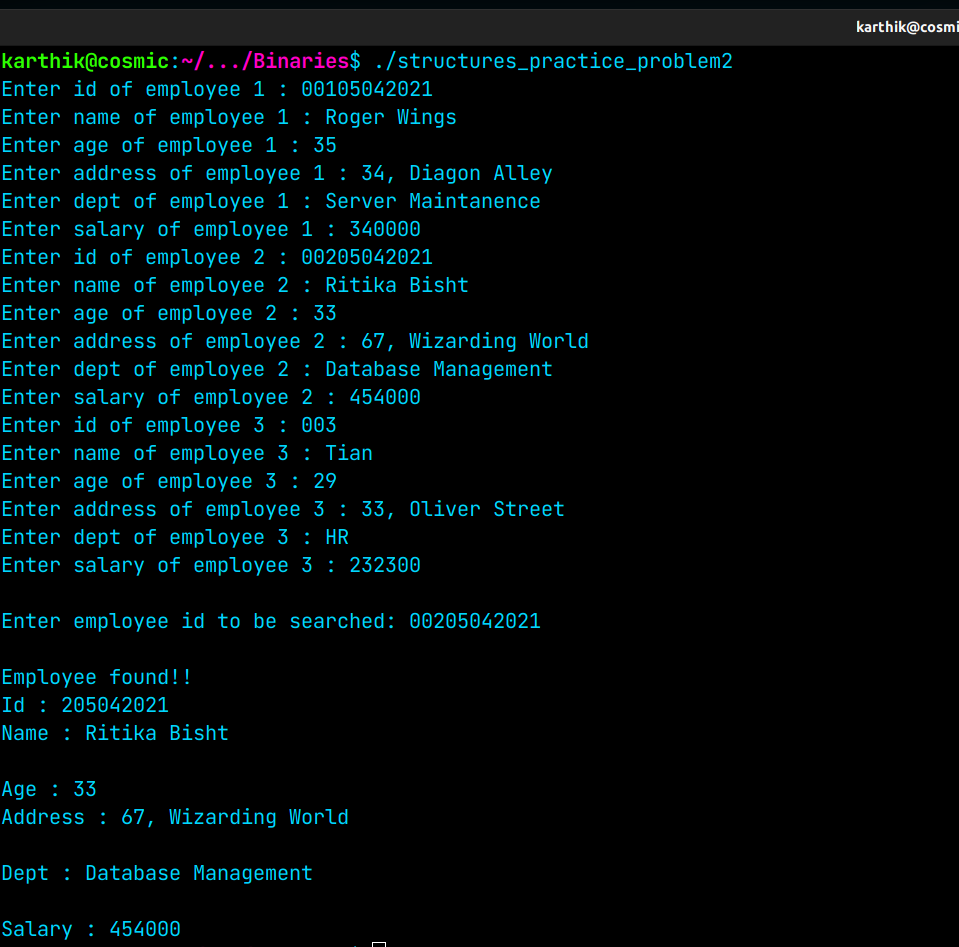
}

}

printf("\nEmployee not found!!");

}

**Output**



**Q10. Write a program to create two files with names EvenFile and OddFile. Input 20 numbers from the user and save even numbers in EvenFile and odd numbers in OddFile.**

**Solution**

/\* Write a program to create two files with names EvenFile and OddFile.

Input 20 numbers from the user and save even numbers in EvenFile and odd numbers in OddFile.\*/

#include <stdio.h>

int main()

{

// Input 20 numbers from the user

int numbers[20], num, num1;

for(int i=0; i<(sizeof(numbers)/sizeof(int)); i++)

{

printf("Enter number %d: ", i+1);

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

}

// Save even numbers in EvenFile and odd numbers in OddFile

FILE \* odd = fopen("OddFile.txt", "w");

FILE \* even = fopen("EvenFile.txt", "w");

for (int i=0; i<(sizeof(numbers)/sizeof(int)); i++)

{

if ((numbers[i]%2)==0)

{

putw(numbers[i], even);

}

else

{

putw(numbers[i], odd);

}

}

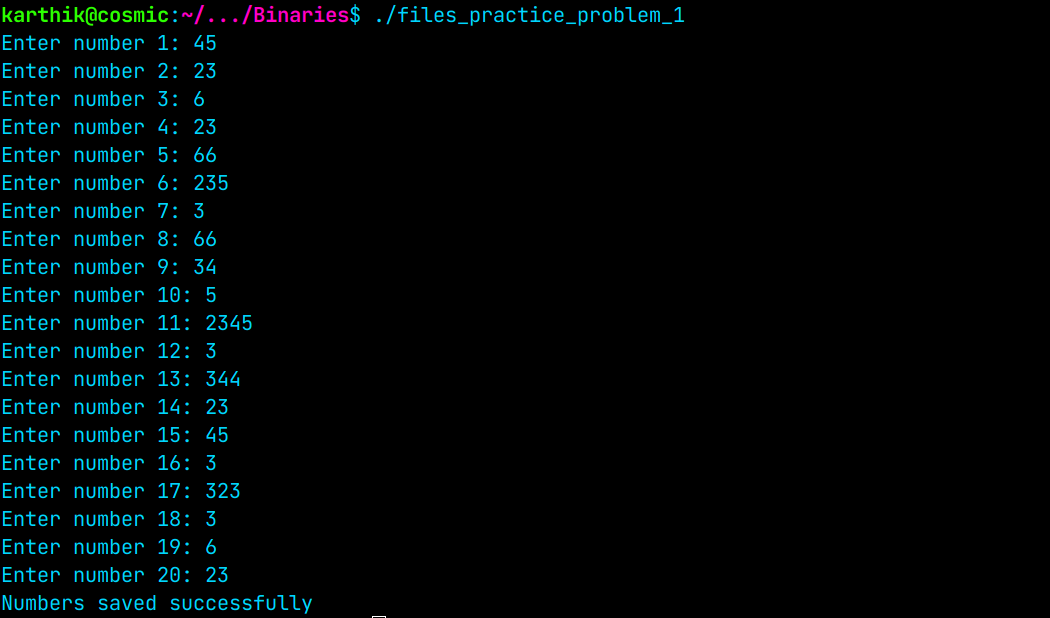
printf("Numbers saved successfully\n");

fclose(odd);

fclose(even);

}

**Output**



**Q11. Write a menu driven program to construct a calculator for following arithmetic operations: addition, subtraction, multiplication, division, average and percentage**

#include <stdio.h>

#include <stdlib.h>

int main()

{

int i, operation, len;

float sum, num, sub, prod, percent;

while (1)

{

printf("\nWhich operation do you want to carry out?\nEnter 1 for addition\n 2 for subtraction\n 3 for multiplication\n 4 for division\n 5 for average\n 6 for percentage \n7 to exit\nEnter here: ");

scanf("%d", &operation);

switch (operation)

{

case 1:

sum=0;

printf("How many Numbers to add?\nEnter Here :");

scanf("%d", &len);

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

{

printf("Enter number %d: ", i+1);

scanf("%f", &num);

sum+=num;

}

printf("sum is %f", sum);

break;

case 2:

printf("How many Numbers to subtract?\nEnter Here :");

scanf("%d", &len);

int first;

printf("Enter number 1: ");

scanf("%d", &first);

sub=first;

for (i=1; i<len; i++)

{

printf("Enter number %d: ", i+1);

scanf("%f", &num);

sub-=num;

}

printf("Result after subtracting is %f.", sub);

break;

case 3:

printf("How many Numbers to take product of?\nEnter Here :");

scanf("%d", &len);

prod=1;

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

{

printf("Enter number %d: ", i+1);

scanf("%f", &num);

prod\*=num;

}

printf("Product is %f.", prod);

break;

case 4:

int div, divs;

printf("Enter dividend: ");

scanf("%d", &div);

printf("Enter divisor: ");

scanf("%d", &divs);

printf("Quotient is %d and remainder is %d\n", (div/divs),(div%divs) );

break;

case 5:

printf("How many Numbers to average?\nEnter Here :");

scanf("%d", &len);

sum=0;

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

{

printf("Enter number %d: ", i+1);

scanf("%f", &num);

sum+=num;

}

printf("Average is %f", sum/len);

break;

case 6:

float n, p;

printf("Enter number and what percentage of it is to be calculated.\n");

printf("Enter number: ");

scanf("%f", &n);

printf("Enter what percentage of it is to be calculated: ");

scanf("%f", &p);

percent=(n/100)\*p;

printf("%f percentage of %f is %f", p, n, percent);

break;

case 7:

printf("Abort!\n");return 0;

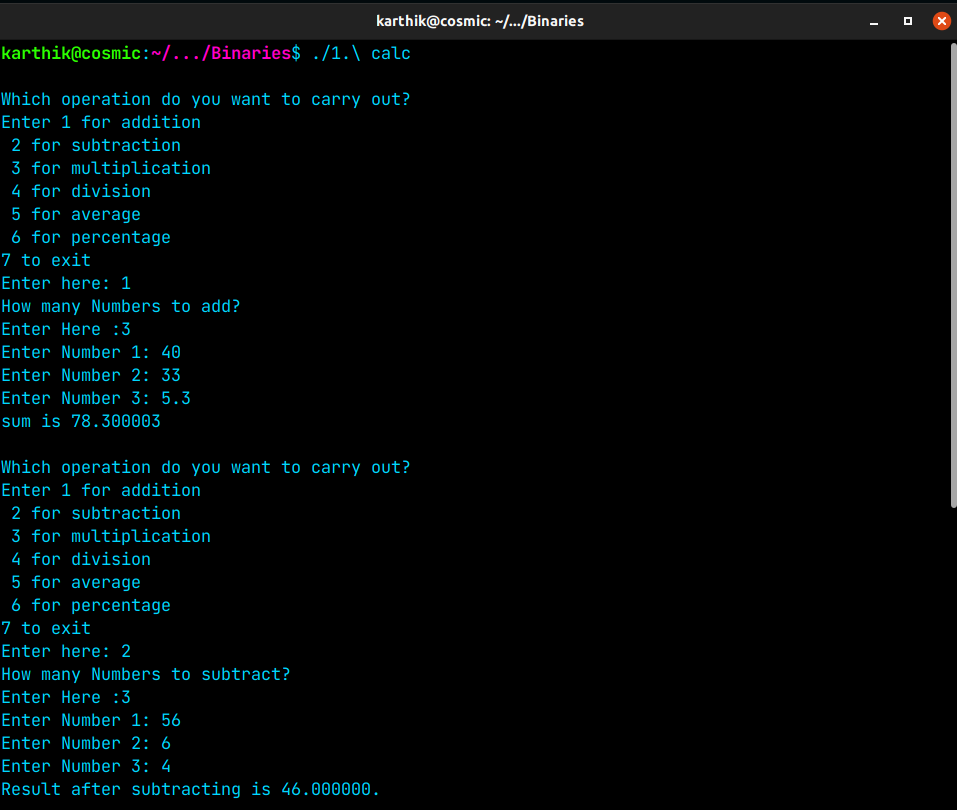
}

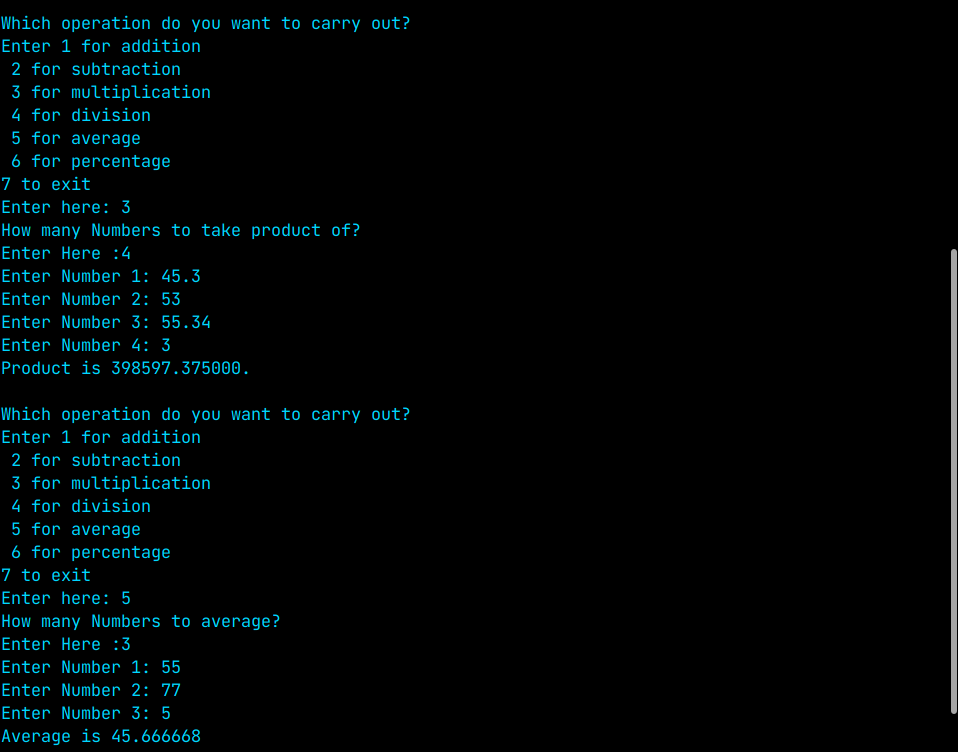
printf("\n");

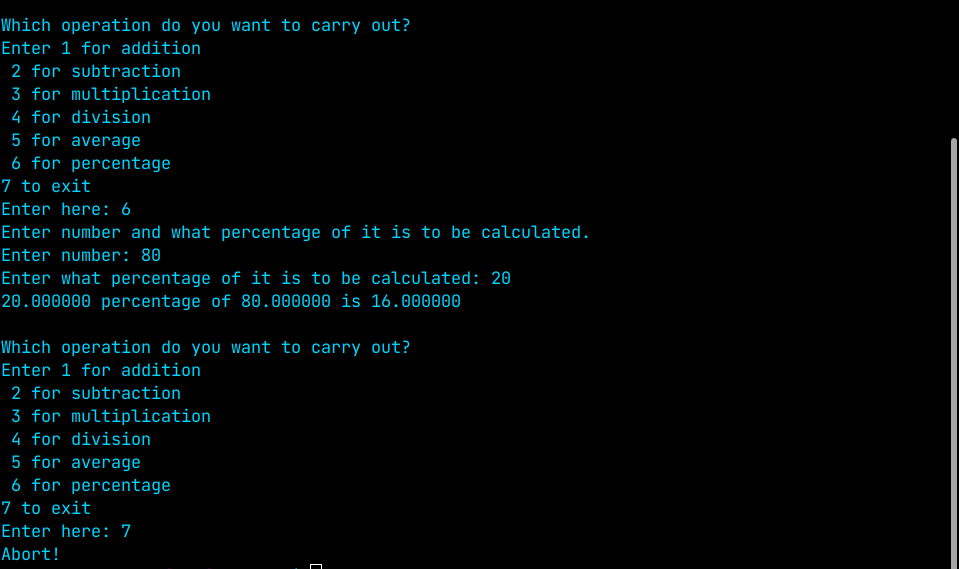
}

}

**Output**



****

****

**Q12. Write a menu driven program to perform the following operations:**

**(i) Print armstrong numbers upto N**

**(ii) Display prime numbers between 1 to N**

**(iii) Reverse of an integer**

**Solution**

#include <stdio.h>

int prime\_num(int number);

int sum\_pow(int number, int digits);

int count\_digits(int number);

int reverse(int number);

int count\_digits(int number)

{

if (number>0)

{

number=number/10;

return 1 + count\_digits(number);

}

else

{

return 0;

}

}

int sum\_pow(int number, int digits)

{

int i, j=digits, digit=1;

if (number>0)

{

for (i=1;1<=j;j--)

{

digit=digit\*(number%10);

}

return digit+ sum\_pow((number/10), digits);

}

else

digit-=1;

return digit;

}

int prime\_num(int number)

{

int i, j=(number/2);

for (i=2;i<=j;i++)

{

if (number%i==0)

{

return 0;

}

}

return 1;

}

int reverse(int number)

{

int num1=0;

while (number>0)

{

num1=(num1\*10)+(number%10);

number/=10;

}

return num1;

}

int main()

{

int n, operation, i,num, og\_num;

while (1)

{

printf("Input which operation is to be performed\n");

printf("1 to print armstrong numbers upto N\n");

printf("2 to print prime numbers between 1 to N\n");

printf("3 to reverse an integer\n");

printf("4 to quit the program\n");

printf("Enter here: ");

scanf("%d", &operation);

switch (operation)

{

case 1:

printf("Enter value of n: ");

scanf("%d", &n);

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

{

int digits, sum;

digits=count\_digits(i);

sum=sum\_pow(i, digits);

if (sum==i)

{

printf("%d, ", i);

}

}

break;

case 2:

printf("Enter N: ");

scanf("%d", &n);

printf("Prime numbers between 1 and %d are: ", n);

for (i=1; i<=n; i++)

{

if(prime\_num(i)==1)

{

printf("%d, ", i);

}

}

break;

case 3:

printf("Enter a number: ");

scanf("%d",&num);

og\_num=num;

num=reverse(num);

printf("Reverse of %d is %d\n", og\_num, num);

break;

case 4:

printf("Abort!\n");

return 0;

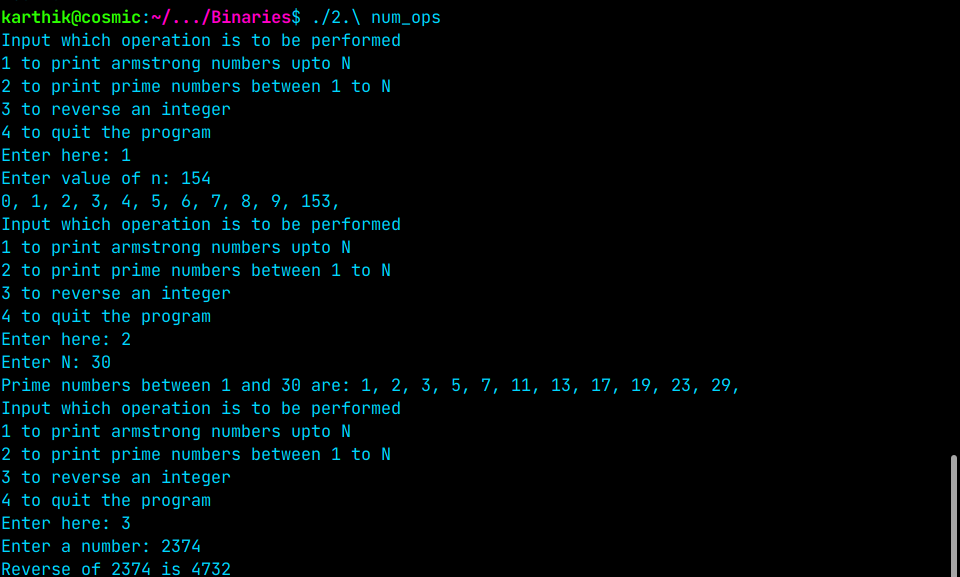
}

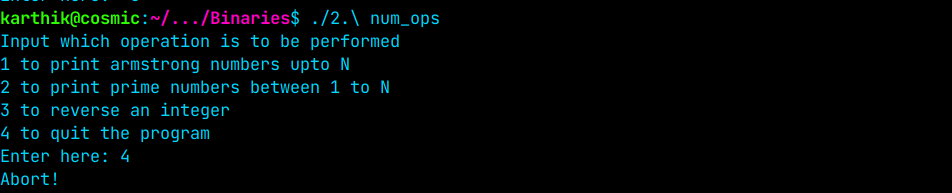
printf("\n");

}

}

**Output**





**Q13. Write a program to convert a hexadecimal number into a binary number**

**Solution**

#include <stdio.h>

#include <string.h>

int main()

{

char num\_in\_hex[100];

printf("Enter a hexadecimal number(shouldn't exceed 100 digits): ");

scanf("%s", num\_in\_hex);

// length of the hexadecimal number is the iterative count

long int iter=strlen(num\_in\_hex);

// store binary equivalent of the hexadecimal number here

// This is better than printing in each iteration since incase a non-hex character homes in between , we can throw an "invalid character message without printing the preceding characters"

char num\_in\_binary[400]="\0";

char invalid\_error[1];

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

{

switch (num\_in\_hex[i])

{

case '0':

strcat(num\_in\_binary,"0000");

break;

case '1':

strcat(num\_in\_binary,"0001");

break;

case '2':

strcat(num\_in\_binary,"0010");

break;

case '3':

strcat(num\_in\_binary,"0011");

break;

case '4':

strcat(num\_in\_binary,"0100");

break;

case '5':

strcat(num\_in\_binary,"0101");

break;

case '6':

strcat(num\_in\_binary,"0110");

break;

case '7':

strcat(num\_in\_binary,"0111");

break;

case '8':

strcat(num\_in\_binary,"1000");

break;

case '9':

strcat(num\_in\_binary,"1001");

break;

case 'A':

case 'a':

strcat(num\_in\_binary,"1010");

break;

case 'B':

case 'b':

strcat(num\_in\_binary,"1011");

break;

case 'C':

case 'c':

strcat(num\_in\_binary,"1100");

break;

case 'D':

case 'd':

strcat(num\_in\_binary,"1101");

break;

case 'E':

case 'e':

strcat(num\_in\_binary,"1110");

break;

case 'F':

case 'f':

strcat(num\_in\_binary,"1111");

break;

default:

invalid\_error[0]=num\_in\_hex[i];

i=iter;break;

}

}

if (invalid\_error[0] != '\0')

{

printf("Character %c is NOT a hex character, aborting!!!\n",invalid\_error[0]);

}

else

{

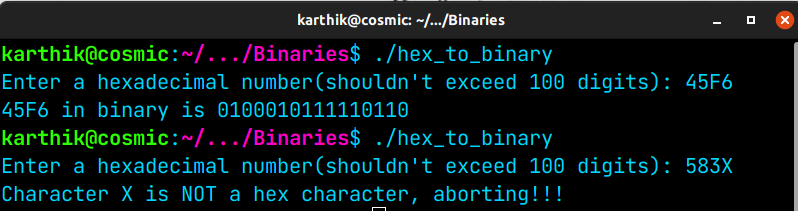
printf("%s in binary is ", num\_in\_hex);

printf("%s\n",num\_in\_binary);

}

}

**Output**



**Q14. Write a program to calculate factorial of a number and display fibonacci series upto N terms using recursive functions**

**Solution**

#include <stdio.h>

int factorial(int n)

{

if (n!=0)

{

return n\*factorial(n-1);

}

else

return 1;

}

void fibonacci(int n)

{

static int n1=0,n2=1,n3;

if(n>0)

{

n3 = n1 + n2;

n1 = n2;

n2 = n3;

printf("%d ",n3);

fibonacci(n-1);

}

}

int main()

{

int choice, num, n;

while (1)

{

printf("Enter \n 1 to calc factorial of a number \n 2 to print fibonacci series upto N terms\n 3 to exit\n Enter here: ");

scanf("%d", &choice);

switch (choice)

{

case 1:

printf("Enter number: ");

scanf("%d", &num);

printf("Factorial is %d\n", factorial(num));

break;

case 2:

printf("Enter N: ");

scanf("%d", &n);

printf("Fibonacci series upto %d terms is \n", n);

printf("0 1 ");fibonacci(n-2);

break;

case 3:

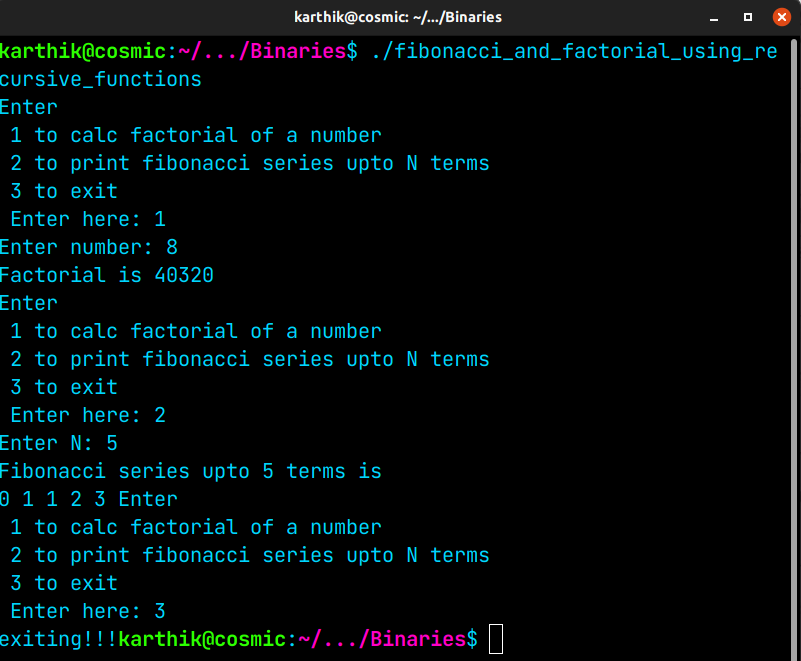
printf("exiting!!!");return 0;

}

}

}

**Output**



**Q15. Write a program to perform (i)matrix addition, (ii)matrix multiplication and (iii) matrix transporse on 2D arrays**

#include <stdio.h>

int main()

{

while(1==1)

{

int operation;

int i, j, k, rows0, cols0, rows1, cols1, matrix0[10][10], matrix1[10][10], matrix2[10][10];

printf("Choose which operation do you want to execute\n");

printf("Enter\n");

printf("1 for matrix Addition\n");

printf("2 for matrix Multiplication\n");

printf("3 for matrix Transpose\n");

printf("4 to exit\n");

printf("Enter here: ");

scanf("%d", &operation);

switch (operation)

{

case 1:

printf("Number of rows and columns shouldn't exceed 10\n");

printf("Enter number of rows: ");

scanf("%d",&rows0);

printf("Enter number of columns: ");

scanf("%d",&cols0);

printf("Input values for First Matrix\n");

for(i=0; i<rows0; i++)

{

printf("Enter row %d with each value separated with space\n", i+1);

for(j=0; j<cols0; j++)

{

scanf("%d", &matrix0[i][j]);

}

}

printf("Input values for second Matrix\n");

for(i=0; i<rows0; i++)

{

printf("Enter row %d with each value separated with space\n", i+1);

for(j=0; j<cols0; j++)

{

scanf("%d", &matrix1[i][j]);

}

}

for(i=0; i<rows0; i++)

{

for(j=0; j<cols0; j++)

{

matrix2[i][j] = matrix0[i][j] + matrix1[i][j];

}

}

printf("Matrix sum is \n");

for(i=0; i < rows0; i++)

{

for(j=0; j<cols0; j++)

{

printf("%d ", matrix2[i][j]);

}

printf("\n");

}

break;

case 2:

printf("Number of Rows and columns shouldn't exceed 10\n");

printf("Enter number of rows and columns for first Matrix\n");

printf("Enter number of rows: ");

scanf("%d", &rows0);

printf("Enter number of columns: ");

scanf("%d", &cols0);

printf("Enter number of rows and columns for second Matrix\n");

printf("Enter number of rows: ");

scanf("%d", &rows1);

printf("Enter number of columns: ");

scanf("%d", &cols1);

if (cols0!=rows1)

{

printf("Number of columns in first matrix should be equal to number of rows in second matrix\n");

return 0;

}

printf("Input values into first Matrix\n");

for(i = 0; i < rows0; i++)

{

printf("Input values in row %d with each value separated with space\n", i);

for(j=0; j< cols0; j++)

{

scanf("%d", &matrix0[i][j]);

}

}

printf("Input values into second Matrix\n");

for(i = 0; i < rows1; i++)

{

printf("Input values in row %d with each value separated with space\n", i);

for(j=0; j< cols1; j++)

{

scanf("%d", &matrix1[i][j]);

}

}

// Matrix multiplication

for (i=0; i<rows0; i++)

{

for (j=0; j< cols1; j++)

{

matrix2[i][j]=0;

for(k=0;k<rows1;k++)

{

matrix2[i][j]+=(matrix0[i][k]\*matrix1[k][j]);

}

}

}

printf("Resultant matrix after matrix multiplication is\n");

for(i=0; i < rows0; i++)

{

for(j=0; j<cols1; j++)

{

printf("%d ", matrix2[i][j]);

}

printf("\n");

}

break;

case 3:

printf("Number of rows and columns shouldn't exceed 10\n");

printf("Enter number of rows: ");

scanf("%d", &rows0);

printf("Enter number of columns: ");

scanf("%d", &cols0);

for(i = 0; i < rows0; i++)

{

printf("Input values in row %d with each value separated with space\n", i);

for(j=0; j< cols0; j++)

{

scanf("%d", &matrix0[i][j]);

}

}

// Transporse the matrix

for (i = 0; i < rows0; i++)

{

for (j=0; j< cols0; j++)

matrix1[j][i] = matrix0[i][j];

}

printf("Resultant matrix after matrix transporse is\n");

for(i=0; i < rows0; i++)

{

for(j=0; j<cols0; j++)

{

printf("%d ", matrix1[i][j]);

}

printf("\n");

}

break;

case 4:

printf("exiting!!!\n");return 0;

}

}

}

**Output**

