**IT161: Introduction to Programming and Problem Solving**

**Lab 4/Assignment 4**

**NAME: SHIDDHARTH BHEEMARAI TASHILDAR  
ROLL NUMBER: 202451151**

**PROGRAMS**

1. **Sum of Natural Numbers using recursion**

**Code:**

#include <stdio.h>

int sum(int *n*){

    if(*n* == 0){return 0;}

    else{

        return *n* + sum(*n*-1);

    }

}

int main(){

    int *num*;

    printf("\nEnter your Number: ");

    scanf("%d",&*num*);

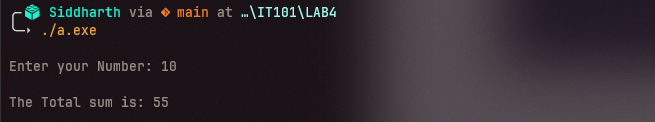
    int *total\_sum* = sum(*num*);

    printf("\nThe Total sum is: %d\n",*total\_sum*);

    return 0;

}

**Output:**

****

1. **Program to check whether a number can be expressed as sum of two prime numbers**

**Code:**

#include <stdio.h>

#include <stdbool.h>

bool isPrime(int *n*){

    int *temp* = 0;

    for(int *i* = 2;*i*<*n*;*i*++){

        if(*n* % *i* == 0){

*temp* = *temp* + 1;

        }

    }

    if(*temp* == 0){

        return true;

    }

    else{

        return false;

    }

}

bool isPossible(int *n*){

    bool *chk* = false;

    for(int *i* = 2; *i* < *n*; *i*++){

        bool *term1* = isPrime(*i*);

        for(int *j* = 2;*j*<*n*;*j*++){

            bool *term2* = isPrime(*j*);

            int *sum* = *i*+*j*;

            if(*term1* && *term2* && *sum* == *n*){

*chk* =  true;

                break;

            }

        }

    }

    if(*chk*){

        return true;

    }

    else{

        return false;

    }

}

int main(){

    int *num*;

    printf("\nEnter your Number: ");

    scanf("%d",&*num*);

    bool *check* = isPossible(*num*);

    if(*check*){

        printf("\nYes it is possible!\n");

        }

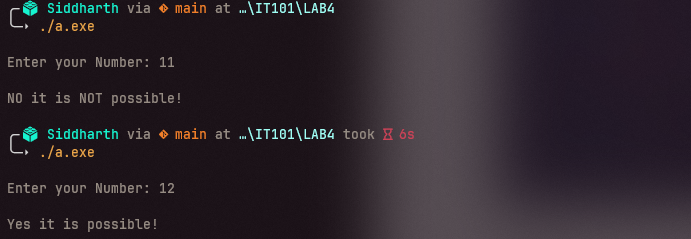
    else{

        printf("\nNO it is NOT possible!\n");

        }

    return 0;

}

**Output:**

1. **Hollow Diamond Pattern**

**Code:**

#include<stdio.h>

int main()

{

  int *n*;

  printf("Enter number of rows: ");

  scanf("%d",&*n*);

  for(int *i*=1; *i*<=*n*; *i*++)

  {

     for(int *j*=*i*; *j*<=*n*; *j*++)

     {

       printf(" ");

     }

     for(int *k*=1; *k*<=2\**i*-1; *k*++)

     {

       if(*k*==1 || *k*==(2\**i*-1)) printf("\*");

       else printf(" ");

     }

     printf("\n");

  }

  for(int *i*=*n*-1; *i*>=1; *i*--)

  {

     for(int *j*=*n*; *j*>=*i*; *j*--)

     {

       printf(" ");

     }

     for(int *k*=1; *k*<=2\**i*-1; *k*++)

     {

       if(*k*==1 || *k*==2\**i*-1) printf("\*");

       else printf(" ");

     }

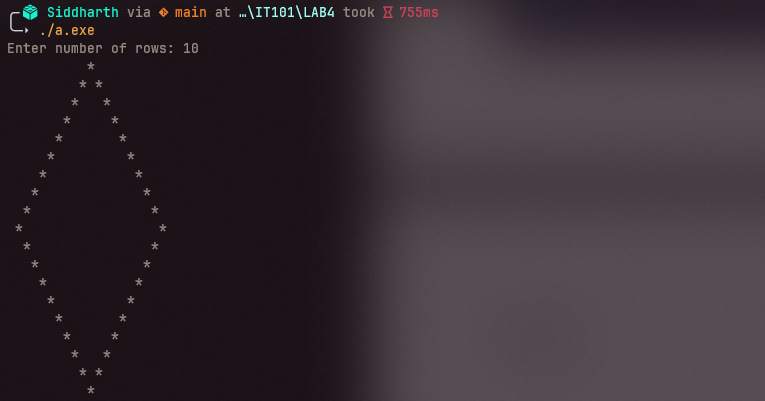
     printf("\n");

  }

  return 0;

}

**Output:**

****

1. **GCD using recursion**

**Code:**

#include <stdio.h>

int gcd(int *a* , int *b*){

    if(*a* == 0 || *b* == 0){

        return *a*+*b*;

    }

    else{

        if(*a* >= *b*){

            return gcd(*a*-*b*,*b*);

        }

        else{

            return gcd(*a*,*b*-*a*);

        }

    }

}

int main(){

    int *num1*;

    int *num2*;

    printf("\nEnter your Numbers: ");

    scanf("%d %d",&*num1*,&*num2*);

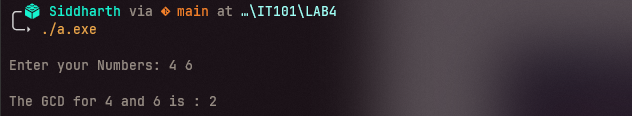
    int *g* = gcd(*num1*,*num2*);

    printf("\nThe GCD for %d and %d is : %d\n",*num1*,*num2*,*g*);

    return 0;

}

**Output:**

****

1. **Count digits of a number using recursion**

**Code:**

#include <stdio.h>

int countDigits(int *n*){

    if(*n* <= 0){

        return 0;

    }

    else{

        return 1 + countDigits(*n*/10);

    }

}

int main(){

    int *num*;

    printf("\nEnter your Number: ");

    scanf("%d",&*num*);

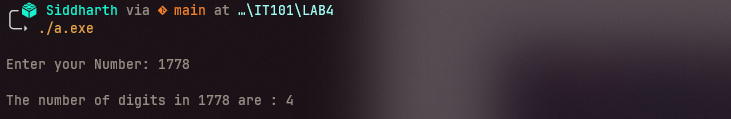
    int *digits* = countDigits(*num*);

    printf("\nThe number of digits in %d are : %d\n",*num*,*digits*);

    return 0;

}

**Output:**

****

**6.Decimal to Binary using recursion**

**Code:**

#include <stdio.h>

int toBinary(int *n*){

    if(*n* <= 0){

        return 0;

    }

    else{

        return *n*%2 + (10 \* toBinary(*n*/2));

    }

}

int main(){

    int *num*;

    printf("\nEnter your Number: ");

    scanf("%d",&*num*);

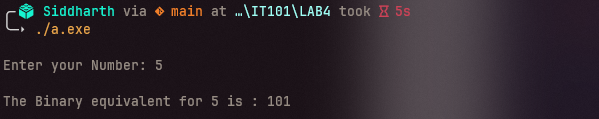
    int *bin* = toBinary(*num*);

    printf("\nThe Binary equivalent for %d is : %d \n",*num*,*bin*);

    return 0;

}

**Output:**

****

**7.Print odd or even numbers in a given range**

**Code:**

#include <stdio.h>

void printOdd(int *a*,int *b*){

    if(*b* <= *a*+2){

        return ;

    }

    else{

        printf(" %d", *a*+2);

        printOdd(*a*+2,*b*);

    }

}

void printEven(int *a*,int *b*){

    if(*b* <= *a*+2){

        return ;

    }

    else{

        printf(" %d", *a*+2);

        printEven(*a*+2,*b*);

    }

}

int main(){

    int *lowerLimit*,*upperLimit*;

    printf("\nEnter your Numbers: ");

    scanf("%d %d",&*lowerLimit*,&*upperLimit*);

    printf("\n Odd Numbers -->  ");

    printOdd(*lowerLimit*,*upperLimit*);

    printf("\n");

    printf("\n Even Numbers -->  ");

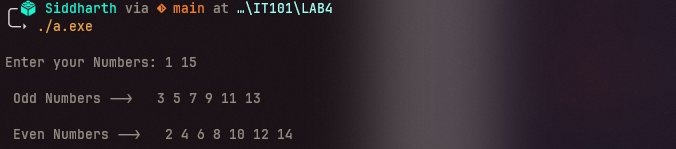
    printOdd(*lowerLimit*-1,*upperLimit*);

    printf("\n");

    return 0;

}

**Output:**

****

**8.Calculate power of any number using recursion**

**Code:**

#include <stdio.h>

int findPower(int *a* , int *b*){

    if(*b* == 0){return 1;}

    else{

        return *a* \* findPower(*a*,*b*-1);

    }

}

int main(){

    int *num1* , *base*;

    printf("\nEnter your numbers: ");

    scanf("%d %d",&*num1*,&*base*);

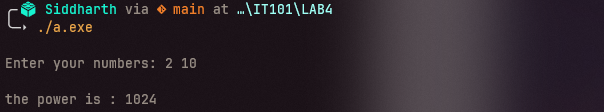
    int *power* = findPower(*num1*,*base*);

    printf("\nthe power is : %d\n",*power*);

    return 0;

}

**Output:**

****