//Question 1:

#include<stdio.h>

int IterativeSum(int i);

int RecursiveSum(int i);

int main(){

int choice=1, sum , i=10;

while(choice!=3){

printf("\n\n1. Choose IterativeSum function to Sum first 10 natural number");

printf("\n2. Choose RecursiveSum function to Sum first 10 natural number");

printf("\n3. Exit");

printf("\n\nEnter your choice: ");

scanf("%d", &choice);

switch(choice){

case 1:

sum = IterativeSum(i);

printf("\n\t\tSum using IterativeSum function is: %d", sum);

break;

case 2:

sum = RecursiveSum(i);

printf("\n\t\tSum using RecursiveSum function is: %d", sum);

break;

default:

return 0;

}

}

}

int IterativeSum(int i){

int j=0;

for(i=1 ; i<=10 ; i++){

j = j + i;

}

return j;

}

int RecursiveSum(int i){

if (i != 0){

return i + RecursiveSum(i - 1);

}

else{

return i;

}

}

//Question 2:

#include<stdio.h>

int RecursiveBinToDec(int i);

int main(){

int choice=1;

int binary, decimal;

while(choice!=3){

printf("\n\n1. Choose IterativeBinToDec function to convert binary to decimal");

printf("\n2. Choose RecursiveBinToDec function to convert binary to decimal");

printf("\n3. Exit");

printf("\n\nEnter your choice: ");

scanf("%d", &choice);

switch(choice){

case 1:

printf("\n\t\tEnter Binary number: ");

scanf("%d", &binary);

decimal = IterativeBinToDec(binary);

printf("\n\t\tDecimal equivalent using IterativeBinToDec function is: %d", decimal);

break;

case 2:

printf("\n\t\tEnter Binary number: ");

scanf("%d", &binary);

decimal = RecursiveBinToDec(binary);

printf("\n\t\tDecimal equivalent using RecursiveBinToDec function is: %d", decimal);

break;

default:

return 0;

}

}

}

int IterativeBinToDec(int i){

int j, z=1, y=1, a, decimal=0;

for(j=i ; j>0 ; j=j/10){

a = j % 10;

if(z==1){

y=y\*1;

}

else{

y=y\*2;

}

decimal = decimal + (a \* y);

z++;

}

return decimal;

}

int RecursiveBinToDec(int i){

if(i==0){

return 0;

}

else{

return (i % 10 + 2 \* RecursiveBinToDec(i / 10));

}

}

//Question 3:

#include<stdio.h>

int Recursive\_Count(int i);

int main(){

int num, occurence;

printf("Enter the number: ");

scanf("%d", &num);

occurence = Recursive\_Count(num);

printf("\nThe count of the occurences is = %d", occurence);

}

int Recursive\_Count(int i){

if(i<1){

return 0;

}

if((i%10)==5 && ((i / 10) % 10)==5){

return(2 + Recursive\_Count(i / 10));

}

else if((i%10)==5){

return(1 + Recursive\_Count(i / 10));

}

else{

return(Recursive\_Count(i / 10));

}

}

//Question 4:

#include<stdio.h>

int fibonacci(int num);

int main(){

int num, n=0, i;

printf("\n\nEnter a number: ");

scanf("%d", &num);

printf("\n\n\nFibonacci sequence:\n");

for(i=1 ; i<=num ; i++){

printf("\t\t\t%d\n", fibonacci(n));

n++;

}

return 0;

}

int fibonacci(int num){

if(num==0 || num==1){

return num;

}

else{

return(fibonacci(num-1) + fibonacci(num-2));

}

}//Question 5:

#include<stdio.h>

int Recursive\_GCD(int num1, int num2);

int main(){

int num1, num2;

printf("\n\nEnter first number: ");

scanf("%d", &num1);

printf("\n\nEnter Second number: ");

scanf("%d", &num2);

printf("\n\nG.C.D of %d and %d is = %d", num1, num2, Recursive\_GCD(num1, num2));

return 0;

}

int Recursive\_GCD(int num1, int num2){

if (num2 != 0){

return Recursive\_GCD(num2, num1%num2);

}

else{

return num1;

}

}

//Question 6:

#include<stdio.h>

int IterativeSum(int i, int j);

int RecursiveSum(int i, int j);

int main(){

int choice=1, sum;

int start\_number, last\_number;

while(choice!=3){

printf("\n\n1. Choose IterativeSum function to calculate sum of range of natural numbers");

printf("\n2. Choose RecursiveSum function to calculate sum of range of natural numbers");

printf("\n3. Exit");

printf("\n\nEnter your choice: ");

scanf("%d", &choice);

switch(choice){

case 1:

printf("\n\t\tEnter the starting number of the range:\t\t");

scanf("%d", &start\_number);

printf("\t\tEnter the last number of the range:\t\t");

scanf("%d", &last\_number);

sum = IterativeSum(start\_number, last\_number);

printf("\n\t\tSum using IterativeSum function is: %d", sum);

break;

case 2:

printf("\n\t\tEnter the starting number of the range:\t\t");

scanf("%d", &start\_number);

printf("\t\tEnter the last number of the range:\t\t");

scanf("%d", &last\_number);

sum = RecursiveSum(start\_number, last\_number);

//For printing the values...

printf("\n\t\t");

for( ; start\_number<=last\_number ; ){

printf("%d ",start\_number);

start\_number++;

if(start\_number>last\_number){

printf("= %d\n",sum);

}

else{

printf("+ ");

}

}

printf("\n\t\tSum using RecursiveSum function is: %d", sum);

break;

default:

return 0;

}

}

}

int IterativeSum(int i, int j){

int total, a;

printf("\n\t\t");

for( ; i<=j ; ){

total = total + i;

printf("%d ",i);

i++;

if(i>j){

printf("= %d\n",total);

}

else{

printf("+ ");

}

}

return total;

}

int RecursiveSum(int i, int j){

if(i==j){

return i;

}

else{

return (i + RecursiveSum(i+1, j));

}

}

//Question 7:

#include<stdio.h>

int IterativeDecToBin(int i);

int RecursiveDecToBin(int i);

int main(){

int choice=1;

int binary, decimal;

while(choice!=3){

printf("\n\n1. Choose IterativeDecToBin function to convert decimal to binary");

printf("\n2. Choose RecursiveDecToBin function to convert decimal to binary");

printf("\n3. Exit");

printf("\n\nEnter your choice: ");

scanf("%d", &choice);

switch(choice){

case 1:

printf("\n\t\tEnter Decimal number: ");

scanf("%d", &decimal);

binary = IterativeDecToBin(decimal);

printf("\n\t\tBinary equivalent using IterativeDecToBin function is: %ld", binary);

break;

case 2:

printf("\n\t\tEnter Decimal number: ");

scanf("%d", &decimal);

binary = RecursiveDecToBin(decimal);

printf("\n\t\tBinary equivalent using RecursiveDecToBin function is: %ld", binary);

break;

default:

return 0;

}

}

}

int IterativeDecToBin(int i){

long binary = 0;

int rem, temp = 1;

while (i!=0){

rem = i%2;

i = i / 2;

binary = binary + (rem \* temp);

temp = temp \* 10;

}

return binary;

}

int RecursiveDecToBin(int i){

if(i==0){

return 0;

}

else{

return ((i % 2) + 10 \* RecursiveDecToBin(i/2));

}

}

//Question 8:

#include<stdio.h>

int Recursive\_Mod(int n, int m, int count);

int main(){

int n, m, result;

int count = 1;

printf("\nEnter the value of Dividend(n): ");

scanf("%d", &n);

printf("\nEnter the value of Divisor(m): ");

scanf("%d", &m);

result = Recursive\_Mod(n, m, count);

printf("\n\n\tThe Final Remainder is = %d", result);

}

int Recursive\_Mod(int n, int m, int count){

if(n==0){

return 0;

}

if(n>=m){

printf("\n\tStep (%d)\t-->\t%d - %d = ", count, n, m);

n=n-m;

printf("%d",n);

count++;

Recursive\_Mod(n, m, count);

}

else{

return n;

}

}