****

**BTC2202 / ICS2014 : Computer Organization and Architecture**

**Assignment #2**

Njenga Dennis Mang’uru - 121047

Wamai Alex Kariuki - 121660

Githuba Jeremy Richard - 115862

Kioko Xavier Kimali - 115059

**QUESTION 2 (A)**

* Coding language used: C++
* Test data numbers include 5, 10,15, 31
* This was the code generated.

#include <iostream>

using namespace std;

int main()

{

int a[10], n, i, z; //Initialization of variable and array to store binary data.

int temp, x=1, j, r;

char hex[50];

cout<<"Enter the number to convert: ";

cin>>n; // Input any number.

z = n;

temp = n;

for(i=0; n>0; i++) //Binary Conversion.

{

a[i]=n%2; //Modulus function to divide the number input by 2 and stores

remainder in array A.

n= n/2; //The loop ends when the quotient becomes 0.

}

cout << "Binary equivalent of " << z << " is : "; //Output binary number.

for(i=i-1 ;i>=0 ;i--) //Hexadecimal Conversion.

{

cout<<a[i];

}

while (temp != 0)

{

r = temp % 16; //num is copied to a temporary variable ‘temp’ and is divided

by 16.

if (r < 10){

hex[x++] = r + 48; // If the remainder is less than 10, 48 is added to it according to

ASCII value and stored in the character array hex.

}

else{

hex[x++] = r + 55; // Else 55 is added to the remainder and stored in hex.

}

temp = temp / 16; //The loop ends when the quotient becomes 0.

}

cout << "\nHexadecimal equivalent of " << z << " is : "; //Output Hexadecimal number

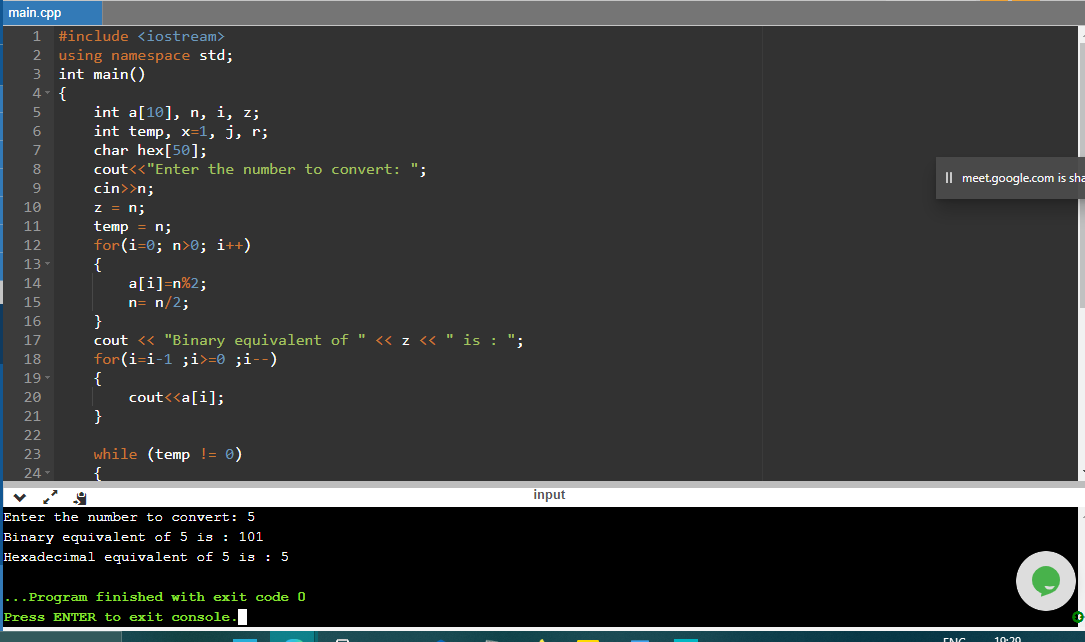
for (j = x; j > 0; j--)

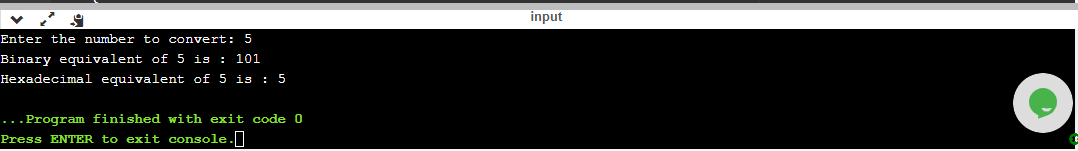
cout << hex[j]; // The array hex is then printed in reverse order which is the

hexadecimal equivalent of the number entered.

}

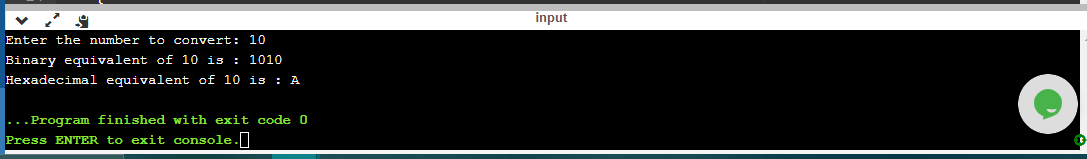
**Test Data Output**

**Input number 5**



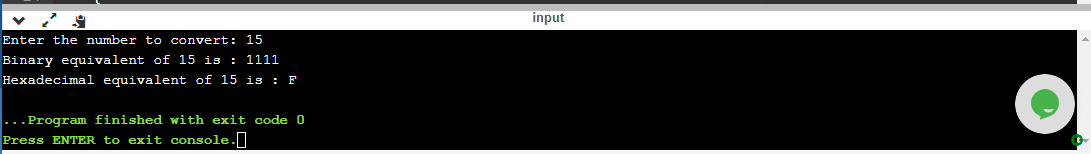
**Input number 10**





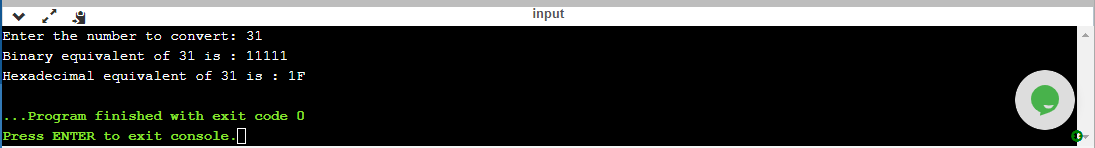
**Input number 15**





**Input number 31**





**QUESTION 2(B)**

import java.text.DecimalFormat;

public class Main

{

public static void main(String[] args) {

double min = 0.0;

double max = 500.0;

int x = 0;

int y = 1;

System.out.print("S/No. \t");

System.out.print("Decimal Number \t");

System.out.print("Binary Number \t \t");

System.out.print("Remarks \n");

do{

int placement = 0;

int place = 0;

int binary [] = new int [10]; //array to store integer binary equivalent

int whole [] = new int [6]; //array to store fraction part binary equivalent

System.out.print(y);

y++; //increments the numbering (1-30)

System.out.print("\t");

double random = min + Math.random() \* (max - min); //generates random number within the range (max-min)

DecimalFormat numberFormat = new DecimalFormat("#.000"); //restricts the number of decimal places for the random number to 3

System.out.print(numberFormat.format(random)+ "\t"); //prints the random number

int intPart = (int) random; //TypeCast the integer part to a number, for easy working

String Decimal = numberFormat.format(random - intPart); //Separates integer part from the fraction part

// System.out.print("Integer Part: " + intPart);

// System.out.print("Decimal Part: " + Decimal);

System.out.print("\t");

x++;

do //converts integer part to binary

{

if(intPart %2 == 0) //divisible by 2?

{

intPart = intPart/2;

binary[placement] = 0;

}

else

{

intPart = intPart/2;

binary[placement] = 1;

}

placement++; //increments the array index

}while(intPart > 0); //process continues until integer is <= 0

double i = Double.parseDouble(Decimal);

//TypeCasts Decimal part of number from String to double

//System.out.println(i);

int count = 0; //counter for the number of iterations for a recurring number (max of 5 decimal places)

do

{

if(i > 0.999) //number starts from 1.000++

{

i = (Double.parseDouble(numberFormat.format( i - (int) i)) \* 2); //takes only the fraction part of the number and multiplies by 2

//System.out.println(i);

whole[place] = (int) i; //stores the 1(one) in an array

place++; //increments array index

count++; //increments the counter for the ,ax decimal places

}

else if (i<=0.999)

{

i = i \* 2; //multiplies fraction by 2

whole[place] = (int) i; //stores the 0(zero) in an array

place++;

count++;

}

}while(count < 6); //process continues until a max of 5 decimal places

for (int r = binary.length - 1; r >= 0; r--) //prints decimal integer's binary array backwards

{

System.out.print(binary[r]);

}

System.out.print(".");

for (int s = 0; s <whole.length - 1 ; s++) //prints the decimal fraction's binary equivalent array backwards

{

System.out.print(whole[s]);

}

System.out.print("\t \t");

//System.out.print(count);

if(count >= 5) //if count >= 5, it means the array for the fraction part is full, thus the remarks apply

{

System.out.print("Approximate");

}

else

{

System.out.print("Exact");

}

System.out.print("\n");

System.out.print("\n");

}while(y<31); //process continues until 30 random numbers are generated

}

}

