Madan Bhandari Memorial College

Experiment 1: Data representation

Objective:

1. To illustrate the concept of data representation

Theory:

Data representation: Data representation in computer architecture refers to how data is stored, processed, and transmitted, primarily in binary form using 0s and 1s. Digital computers store and process information in binary form as digital logic has only two values "1" and "0" or in other words "True or False" or also said as "ON or OFF". There are several ways for data representation some of the most important types include:

- **Binary (Base 2):** This uses digits 0 and 1. Computers internally represent all data in binary format. For example, the number 2 is represented as 10 in binary.
- **Octal (Base 8):** It uses digits from 0 to 7. An example of an octal number is 324017.
- **Decimal (Base 10):** The standard number system used in daily life, which includes digits 0 to 9. An example is 875629.
- **Hexadecimal (Base 16):** It uses digits 0-9 and letters A-F, where A represents 10, B represents 11, and so on. Hexadecimal numbers are often used in programming and digital systems.

Algorithm:

i. Start

ii. Initialize a function to find binary 1's and 2's complement, hexadecimal, octal equivalent of number

iii. Input a number to be converted

iv. Find the equivalent numbers

v. Print the binary 1's and 2's complement, hexadecimal and octal equivalent numbers.

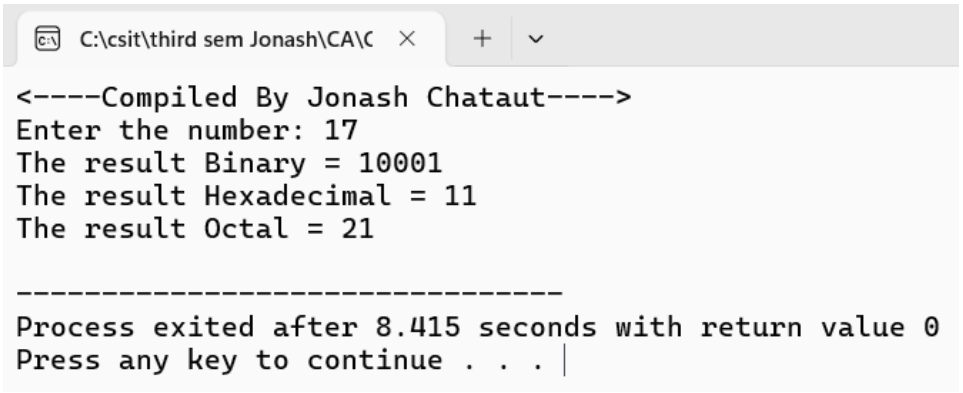
vi. Stop

Source Code:

```
#include <stdio.h>
int conversion(int num, int base)
{
  int rem;
  if (num == 0)
  {
```

```
    return 1;
  }
  else
  {
    rem = num % base;
    conversion(num / base, base);
    if (base == 16 && rem >= 10)
    {
      printf("%c", rem + 55);
    }
    else
    {
      printf("%d", rem);
    }
  }
}
int main()
{
  int num;
  printf("<----Compiled By Jonash Chataut---->\n");
  printf("Enter the number: ");
  scanf("%d", &num);
  if (num != 0)
  {
    printf("The result Binary = ");
    conversion(num, 2);
    printf("\n");
    printf("The result Hexadecimal = ");
    conversion(num, 16);
    printf("\n");
    printf("The result Octal = ");
    conversion(num, 8);
    printf("\n");
  }
  else
  {
    printf("The  binary number is=O\n");
    printf("The  hexadecimal number is=O\n");
    printf("The  octal number is=O\n");
  }
}
```

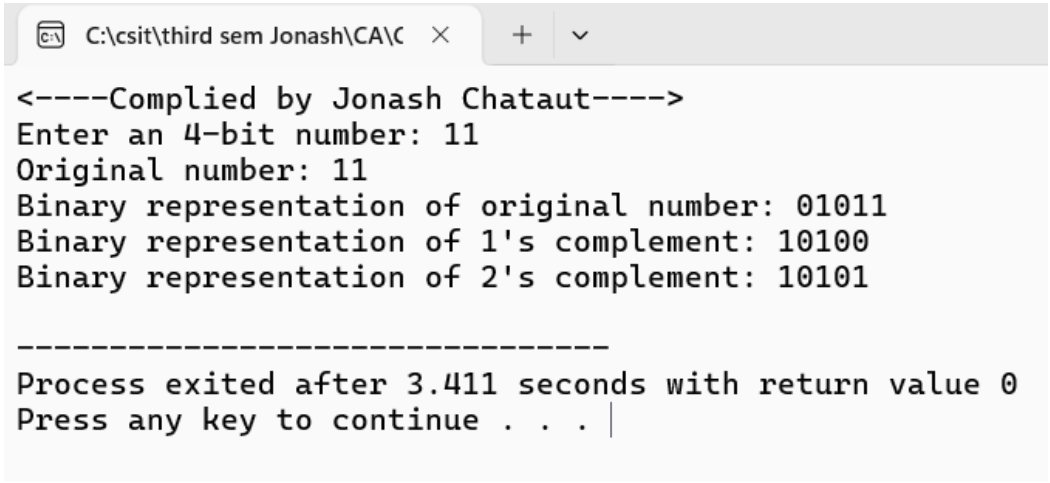
Output:



Source Code:

```
#include <stdio.h>
int i;
char onesComplement(char num)
{
   return ~num;
}
char twosComplement(char num)
{
   return onesComplement(num) + 1;
}
void printBinary(char num)
{
   for (i = 4; i >= 0; i--)
   {
      printf("%d", (num >> i) & 1);
   }
}
int main()
{
   char num;
   printf("<----Complied by Jonash Chataut---->\n");
   printf("Enter an 4-bit number: ");
   scanf("%d", &num);
   unsigned char onesComp = onesComplement(num);
   unsigned char twosComp = twosComplement(num);
   printf("Original number: %d\n", num);
   printf("Binary representation of original number: ");
   printBinary(num);
   printf("\n");
   printf("Binary representation of 1's complement: ");
   printBinary(onesComp);
   printf("\n");
   printf("Binary representation of 2's complement: ");
   printBinary(twosComp);
   printf("\n");
   return 0;
}
```

Output:


C:\csit\third sem Jonash\CA\C
<----Complied by Jonash Chataut---->
Enter an 4-bit number: 11
Original number: 11
Binary representation of original number: 01011
Binary representation of 1's complement: 10100
Binary representation of 2's complement: 10101

Process exited after 3.411 seconds with return value 0
Press any key to continue . . .