

Assignment Number

2

Problem Statement

Program in C to convert a number from any base to any base.

Theory

A number or quantity which is arbitrarily made the fundamental number of a system; also called base 10 is the base number of Decimal system, 2 is the base number of the binary system, 8 is the base number of octal system & 16 is the base number of the hexadecimal system of numeration.

The base of a number system is that number, raised to the zero power, gives the lowest positional value; raised to the 1st power, gives the 2nd positional value and so on. Base conversion is a process to convert a number which can be of binary or octal or decimal or hexadecimal to any of these four bases. To implement this program we have to carefully examine the base of conversion. Complete knowledge of the bases is needed to perform this program. Example:-

Decimal to Binary conversion :- An easy method of converting a decimal number into a binary number is by dividing the decimal number by 2 respectively, until the

quotient of zero is obtained. The binary number is obtained by taking the remainder after each division in the reverse order. The decimal number 21 is equivalent to 10101.

Decimal to Octal conversion :- The octal equivalent of a decimal number can be obtained by dividing a given decimal number by 8 repeatedly, until a quotient of 0 is obtained. The decimal number 1071 is equivalent to octal 2057.

Decimal to Hexadecimal conversion :- The hexadecimal equivalent of a decimal number can be obtained by dividing a given decimal number by 16 repeatedly, until a quotient of 0 is obtained. The decimal number 431 is equivalent to hexadecimal 1AF.

Algorithm

Variable listing :

Variable name	Purpose
ch	Used for checking the condition in do - while loop.
n	Number to be entered by the user.
x[]	Stores each digit of the input number after converting it to decimal.
ibase	Stores the input base.
obase	Stores the output base.
c	Stores the output.
p	Stores the value of i.
hex[]	String to be entered by the user.
y[]	Gives the remainder.
decimal	Gives the decimal number.
value	Used to calculate decimal.
length	Stores the length of the string hex.
i	Counter variable used in for loop.

Algorithm for method main() :-

[Starting of do - while loop]

Step 1 : Set $i = 1$

Step 2 : Input : "Enter the input base : "

Step 3 : Read ibase

Step 4 : If (ibase \neq 16) then

a. Input : "Enter the number : "

b. Read n

Step 5 : Else

a. Input : "Enter the number : "

b. Read hex

[End of if - else structure]

Step 6 : Input : "Enter the output base : "

Step 7 : Read obase

Step 8 : If (ibase \neq 10) then

a. If (ibase \neq 16) then

i. Set $c =$ call the method convert10(n , ibase)

b. Else

i. Set $c =$ call the method convert16to10(hex , ibase)

[End of inner if - else structure]

Step 9 : Else

a. Set $c = n$

[End of outer if - else structure]

Step 10 : If (obase = 10) then

a. Print "The output is : ",c

[End of if structure]

Step 11 : Repeat through step a to step d while ($c \geq 1$) do

a. Set $x[i] = c \text{ MOD } obase$

b. Set $c = c / obase$

c. Set $p = i$

d. Set $i = i + 1$

[End of while loop]

Step 12 : If (obase \neq 10) then

```

a. Print "The output is : "
b. Repeat through step (i) to step (iii) for ( i = p to 1 )
    i. If ( x[i] >= 10 ) then
        1. Call the method base16(a , l)
    ii. Else
        1. Print x[i]
    [End of inner if – else structure]
    iii. Set i = i – 1
[End of for loop]
[End of outer if – else structure]
Step 13 :      Input : "Do you want to continue(Y/N)? : "
Step 14 :      Read ch
Step 15 :      If ( ch = 'n' OR ch = 'N') then
    a. Go to step 18
[End of if structure]
Step 16 :      Repeat through step 1 to step 37 while ( ch = 'y' OR ch =
'Y')
Step 17 :      do
[End of do - while loop]
Step 18 :      Stop
[End of method main()]

```

Algorithm for method convert10(n , ibase) :-

```

Step 1 : Set i = 0
Step 2 : Set s = 0
Step 3 : Repeat through step a to step d while (n >= 1) do
    a. Set y[i] = n MOD 10
    b. Set n = n / 10
    c. Set p = l
    d. Set i = i + 1
[End of while loop]
Step 4 : Repeat through step a to step b for (i = 0 to p)
    a. Set s = s + y[i] * ibasei

```

b. Next I
[End of for loop]
Step 5 : Return s
Step 6 : Stop
[End of method convert10()]

Algorithm for method convert16to10(hex[20] , ibase) :-

Step 1 : Set decimal = 0
Step 2 : Set i = 0
Step 3 : Find the length of the string hex and store it into length
Step 4 : Set length = length - 1
Step 5 : Repeat through step 6 to step 40 for (i = 0 to hex[i] ≠ NULL
Step 6 : [Starting of switch case block]
Step 7 : Check whether hex[i] = 0 then go to step 7 else go to step 8
 a. i) Set value = 0 // case 0
 b. ii) Go to step 23
Step 8 : Check whether hex[i] = 1 then go to step 9 else go to step 10
 a. i) Set value = 1 // case 1
 b. ii) Go to step 23
Step 9 : Check whether hex[i] = 3 then go to step 11 else go to step 12
 a. Set value = 2 // case 2
 b. Go to step 23
Step 10 : Check whether hex[i] = 4 then go to step 11 else go to step 12
 a. Set value = 3 // case 3
 b. Go to step 23
Step 11 : Check whether hex[i] = 4 then go to step 11 else go to step 12

- a. Set value = 4 // case 4
- b. Go to step 23
- Step 12 : Check whether $\text{hex}[i] = 5$ then go to step 11 else go to step 12
 - a. Set value = 5// case 5
 - b. Go to step 23
- Step 13 : Check whether $\text{hex}[i] = 6$ then go to step 11 else go to step 12
 - a. Set value = 6// case 6
 - b. Go to step 23
- Step 14 : Check whether $\text{hex}[i] = 7$ then go to step 11 else go to step 12
 - a. Set value = 7// case 7
 - Go to step 23
- Step 15 : Check whether $\text{hex}[i] = 8$ then go to step 11 else go to step 12
 - a. Set value = 8// case 8
 - b. Go to step 23
- Step 16 : Check whether $\text{hex}[i] = 9$ then go to step 11 else go to step 12
 - a. Set value = 9// case 9
 - b. Go to step 23
- Step 17 : Check whether $\text{hex}[i] = a$ then go to step 11 else go to step 12
 - a. Set value = 10 // case 10
 - b. Go to step 23
- Step 18 : Check whether $\text{hex}[i] = b$ then go to step 11 else go to step 12
 - a. Set value = 11 // case 11
 - b. Go to step 23

Step 19 : Check whether $\text{hex}[i] = c$ then go to step 11 else go to step 12

a. Set value = 12 // case 12

b. Go to step 23

Step 20 : Check whether $\text{hex}[i] = d$ then go to step 11 else go to step 12

a. Set value = 13 // case 13

b. Go to step 23

Step 21 : Check whether $\text{hex}[i] = e$ then go to step 11 else go to step 12

a. Set value = 14 // case 14

b. Go to step 23

Step 22 : Check whether $\text{hex}[i] = f$ then go to step 11 else go to step 12

a. Set value = 15 // case 15

b. Go to step 23

[End of switch case block]

Step 23 : Set $\text{decimal} = \text{decimal} + \text{value} * \text{ibase}^{\text{length}}$

Step 24 : Set $\text{length} = \text{length} - 1$

Step 25 : Next I

[End of for loop]

Step 26 : Return decimal

Step 27 : Stop

[End of method convert16to10()]

Algorithm for method base16(x[25] , i) :-

Step 1 : Set $c = x[i] + 55$

Step 2 : Print c

Step 3 : Stop

[End of method base16()]

Source code

```
#include<stdio.h>
#include<conio.h>
#include<stdlib.h>
#include<string.h>
#include<math.h>

long int convert10(long int,long int);
void base16(long int x[25],long int);
long int convert16to10(char hex[20],long int);
int l;
char ch;

void main()
{
    do
    {
        long int n,x[25],ibase,obase,c,p,i=1;
        char hex[17];
        printf("\n Enter the input base: ");
        scanf("%ld",&ibase);
        if(ib!=16)
        {
            printf("\n Enter the number : ");
            scanf("%ld",&n);
        }
        else
        {
            printf("\n Enter the number : ");
            fflush(stdin);
            gets(hex);
        }
        printf("\n Enter the output base : ");
        scanf("%ld",&obase);
        if(ibase!=10)
        {
```



```

        if(ibase!=16)
            c=convert10(n,ibase);
        else
            c=convert16to10(hex,ibase);
    }
    else
        c=n;
    if(obase==10)
    {
        printf("\n The output is : %ld",c);
    }
    while(c>=1)
    {
        x[i]=c%obase;
        c=c/obase;
        p=i;
        i++;
    }
    if(obase!=10)
    {
        printf("\n The output is : ");
        for(i=p;i>=1;i--)
        {
            if(x[i]>=10)
                base16(a,i);
            else
                printf("%ld",x[i]);
        }
    }
    printf("\n\n Do you want to continue(Y/N)? : ");
    fflush(stdin);
    scanf(" %c",&ch);
    if(ch=='n' || ch=='N')
        exit(0);
}while(ch=='y' || ch=='Y');
}

long int convert10(long int n,long int ibase)
{
    long int y[25],i=0,s=0,p,decimal,value;

```

```
while(n>=1)
{
    y[i]=n%10;
    n=n/10;
    p=i;
    i++;
}
for(i=0;i<=p;i++)
    s=s+y[i]*pow(ibase,i);
return s;
}

long int convert16to10(char hex[20],long int ibase)
{
    long int length,value,decimal=0;
    int i=0;
    length=strlen(hex);
    length--;
    for(i=0;hex[i]!='\0';i++)
    {
        switch(hex[i])
        {
            case '0': value=0;
            break;
            case '1': value=1;
            break;
            case '2': value=2;
            break;
            case '3': value=3;
            break;
            case '4': value=4;
            break;
            case '5': value=5;
            break;
            case '6': value=6;
            break;
            case '7': value=7;
            break;
            case '8': value=8;
            break;
```

```
        case '9': value=9;
        break;
        case 'A':
        case 'a': value=10;
        break;
        case 'B':
        case 'b': value=11;
        break;
        case 'C':
        case 'c': value=12;
        break;
        case 'D':
        case 'd': value=13;
        break;
        case 'E':
        case 'e': value=14;
        break;
        case 'F':
        case 'f': value=15;
        break;
    }
    decimal=decimal+value*pow(ibase,length);
    length--;
}
return decimal;
}

void base16(long int x[25], long int i)
{
    char c;
    c=x[i]+55;
    printf("%c",c);
}
```

Input and Output

Set 1 :-

Enter the input base: 10

Enter the number : 192

Enter the output base : 2

The output is : 11000000

Do you want to continue(Y/N)? : Y

Set 2 :-

Enter the input base: 10

Enter the number : 37

Enter the output base : 8

The output is : 45

Do you want to continue(Y/N)? : Y

Set 3 :-

Enter the input base: 10

Enter the number : 987

Enter the output base : 16

The output is : 3DB

Do you want to continue(Y/N)? : Y

Set 4 :-

Enter the input base: 2

Enter the number : 1011100011

Enter the output base : 8

The output is : 1343

Do you want to continue(Y/N)? : Y

Set 5 :-

Enter the input base: 2

Enter the number : 1000010101

Enter the output base : 10

The output is : 533

Do you want to continue(Y/N)? : Y

Set 6 :-

Enter the input base: 2

Enter the number : 1110100111

Enter the output base : 16

The output is : 3A7

Do you want to continue(Y/N)? : Y

Set 7 :-

Enter the input base: 8

Enter the number : 7476

Enter the output base : 2

The output is : 111100111110

Do you want to continue(Y/N)? : Y

Set 8 :-

Enter the input base: 16

Enter the number : AE5C7D

Enter the output base : 10

The output is : 11426941

Do you want to continue(Y/N)? : N

Discussion

1. To convert a number from one base to another one must enter two different bases.
2. If the entered bases are same then the output will be same as like the input. So for better programming one must check at first whether the bases are same or not.
3. If one enters the negative number for bases and also for number of that corresponding base, then user cannot get the desired output. So to get the desired output one must enter the positive number for bases and for numbers.