**4. Topic – Conversion of Binary string to Decimal integer using C program**

* **Problem Statement**

Write a program to do the following:

Read a string of 0's and 1's of length n and print the decimal equivalent of the string treated as an unsigned integer in the binary representation.

For simplicity, assume that the length of the input string is less than or equal to 32. Count and print the number of multiplications and additions used.

**Input example:** /\* Here user will give a binary sring of length <=32 \*/

Enter the binary number

1111111111111111111100

**Output example:**

The decimal equivalent of the given binary string number is 4194300

The number of multiplications used here are 231

The number of additions used here are 21

* **Proposed C Code**

**/\* ---------- binarytodecimal.c--------------- \*/**

#include<stdio.h>

#include<stdlib.h>

#include<string.h>

#include<math.h>

int main()

{

char\* binarystring = (char\*)calloc(33,sizeof(char));

/\* The size of the string is taken 33(considering ‘\0’) as maximum length will be 32.The string will only contain charcters '0' and '1' \*/

printf("Enter the binary number\n");

scanf("%s",binarystring);

int length = strlen(binarystring);

/\* Here we are finding the length of the string \*/

unsigned long long int power =0,sum = 0;

/\*Here unsigned long long is taken for big powers of 2 and also sum of same datatype to store it in each iterations.\*/

int multi\_count = 0;

/\* Here we declare the variable for ounting number of multiplications. \*/

for( int i = length-1 , j = 0 ; i >=0 ; i-- , j++ )

{

/\* The idea of the program is to add successive integers generated after the multiplication of jth power of 2(starting from 0) with the ith character or digit in the string starting from the end.\*/

power = pow(2,j);

sum = sum + power \* (binarystring[i]-'0');

multi\_count = multi\_count + j;

/\* If we multipy 2^3 \*1 means we are performing 2\*2\*2\*1 i.e total 3 multiplicatons in iteration 3.So to count the no of multiplications we just add j in every jth iteration to the multi\_count variable.We just add only once in every iteration so it will be length-1 where length variable is length of the string \*/

}

printf("The decimal equivalent of the given binary string number is %llu\n",sum);

printf("The number of multiplications used here are %d\n",multi\_count);

printf("The number of additions used here are %d\n",length-1);

return 0;

}

**/\*------------------------------------------------------------------------------------------------------------------------- \*/**

* **Conclusion**

**The proposed algorithm has a runtime of O(n) where n is generally the input string size under consideration.**

* **Limitations : As per the question,the maximum string length is 32 so if we give all 32 characters ‘1’ then we will get maximum decimal integer as 2^32 -1. We are considering only unsigned integers also as per the question.**
* **Assumptions: Here we are considering the string length to be less than 32 as per the question.**