BINARY NUMBER GAME

A PROJECT REPORT

SUBMITTED IN COMPLETE FULFILMENT OF THE
REQUIREMENTS FOR THE AWARD OF DEGREE
OF
BACHELOR OF TECHNOLOGY
IN
MATHEMATICS AND COMPUTING

SUBMITTED BY:
PARN PRATAP SINGH
(2K20/B15/64)
UNDER THE SUPERVISION OF:
DR. BINDU VERMA

MATHEMATICS AND COMPUTING
DELHI TECHNOLOGICAL UNIVERSITY
(FORMERLY DELHI COLLEGE OF ENGINEERING)
BAWANA ROAD, DELHI – 110042
MARCH 2021

INTRODUCTION TO THE PROGRAM

I have developed a game using C which is about converting binary to decimal numbers and vice versa. It generates random binary numbers and random decimal numbers and then asks user to input the correct equivalent. It also generates these two combinations randomly. It has two modes, easy and hard. The easy mode has powers of two written on top which helps to calculate the answer quickly and when the user plays the easy mode sufficient times, he can then imagine those number written on top which are not in the hard mode. In the end the score is shown out of 5, and the time also which the user took to answer all the questions. The game also provides instructions which are basically some tricks to get the answer quicker.

This program has 4 user defined functions:- first is the int playeasy() which is called when the user wants to play the easy mode and next is the int playhard() which is called when the user wants to play the hard mode. Then comes two more functions, int checkbintodec() and int checkdectobin() which are used as we can figure out by their name to check if the user has input the correct answer. If not, it also displays what the correct answer is. Both of these functions are called in int playeasy() and int playhard(). Further rand() function is used to generate pseudorandom numbers and srand() is used to change the seed value that is basically changing the initial value for rand() function so as to get different random numbers everytime we play the game.

The idea of this game is original and each line of the code is written by me. If this game helps people to develop a better understanding of binary numbers or atleast help them in inter-conversion of base-2 and base-10 numbers, I would like to take this idea to a next level by adding more bases such as octal and hexadecimal and add more tricks for interconversion such as the 421 rule and the 8421 rule. I would be thankful for any suggestions for improvement.

THE CODE FOR THE PROGRAM

THIS IS THE WHOLE CODE:-

```
#include <stdio.h>
#include <stdlib.h>
#include<time.h>

int randbin[10],deceq,bineq,randdec,wins=0;
int checkbintodec();
int checkdectobin();
int playeasy();
int playhard();
int main()
{   system("color 3b");
    printf("WELCOME TO THE BINARY NUMBER GAME\n\n");
    int i=1,eh;
    choose:
    printf("PRESS 1 FOR EASY MODE , 2 FOR HARD MODE AND 3 FOR
INSTRUCTIONS:");
    scanf("%d",&eh);
    clock_t start,end;
    double time=0;
    start=clock();
    if (eh==1)
    {
              while(i<=5)
                {
                   playeasy();
                i++;
                printf("\n\n\n");


       }
    printf("YOUR SCORE IS %d / 5",wins);
   }
   else if (eh==2)
   {
           while(i<=5)
             {
               playhard();


                i++;
                printf("\n\n\n");


       }
    printf("YOUR SCORE IS %d / 5",wins);
   }
   else if (eh==3)
          {
          printf("FOR EASY MODE:-\nFor binary to decimal- just add the numbers corresponding to 1 written above to calculate the decimal equivalent.");
          printf("\nFor decimal to binary- find the number just lower than the decimal number and write 1 in its place. Now subtract this from the original number and repeat the steps. Insert 0's in between. ");
          printf("\n\nFOR HARD MODE:-\n Now imagine the numbers to be written above as in easy mode and proceed in the same way.\n");
          goto choose;
          }
   else
      {
          printf("INVALID MODE\n");
```

```
            goto choose;
               }
            end=clock();
            time=((double) (end - start)) / CLOCKS_PER_SEC;
            printf("\nTime taken to complete : %f seconds",time);
}
int playeasy()
{
    srand(time(0));
    int x=rand()%2;

    if (x==0)
{
    for(int i=1;i<=6;i++)
    {
            randbin[i]=rand()%2;
            }
            printf("   32   16   8    4    2    1\n");
            printf(" ==============================\n");
            printf(" ");
            for(int i=1;i<=6;i++)
            {
                        printf("| %d |",randbin[i]);
            }
            printf("\n ==============================");
            printf("\nENTER THE DECIMAL EQUIVALENT:");
            scanf("%d",&deceq);
    checkbintodec();
}
    else if (x==1)
```

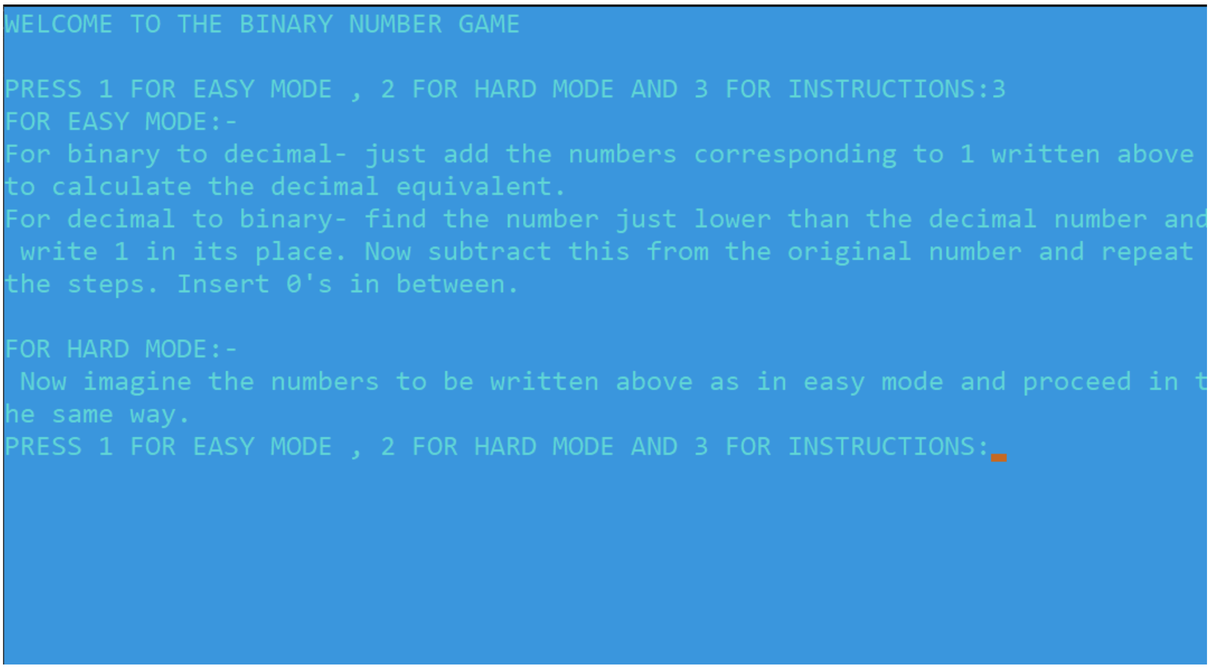
```
 {

   randdec=rand()%64;
   printf("---------------| %d |---------------\n\n",randdec);
   printf("   32   16   8    4    2    1\n");
         printf(" ==============================\n");
         printf(" ");
         for(int i=1;i<=6;i++)
         {
                 printf("| 0 |");
         }
         printf("\n ==============================");
         printf("\nENTER THE BINARY EQUIVALENT:");
         scanf("%d",&bineq);
   checkdectobin();
}
}

int playhard()
{
   srand(time(0));
    int x=rand()%2;

   if (x==0)
{
   for(int i=1;i<=6;i++)
   {
         randbin[i]=rand()%2;
         }
```

```
            printf(" ==============================\n");
            printf(" ");
            for(int j=1;j<=6;j++)
            {
                        printf("| %d |",randbin[j]);
            }
            printf("\n ==============================");
            printf("\nENTER THE DECIMAL EQUIVALENT:");
            scanf("%d",&deceq);
            checkbintodec();
}
 else if (x==1)

 {
             randdec=rand()%65;
    printf("---------------| %d |---------------\n\n",randdec);

            printf(" ==============================\n");
            printf(" ");
            for(int i=1;i<=6;i++)
            {
                        printf("| 0 |");
            }
            printf("\n ==============================");
            printf("\nENTER THE BINARY EQUIVALENT:");
            scanf("%d",&bineq);
    checkdectobin();
 }
}
```

```
int checkbintodec()
{
            int sum=0,base=1;
            for(int i=6;i>=1;i--)
    {
            sum+=(randbin[i])*base;
            base*=2;
            }
            if (sum==deceq)
            {
            printf("CORRECT ANSWER    WELL DONE");
            wins=wins+1;
    }
            else
            {
            printf("WRONG ANSWER    ");
            printf("CORRECT ANSWER IS %d",sum);
    }

}

int checkdectobin()
{
     int checkarr1[10],checkarr2[10];

    int i=0,x,base=1,sum=0;
    while(randdec>0)
          {
             checkarr1[i]=randdec%2;
             randdec/=2;
             i++;
                    }

                    x=i;
                    for(int j=0;j<x;j++)
                    {
                         checkarr2[j]=checkarr1[i-1];
                         i--;
                    }
            for (int i=x-1;i>=0;i--)
            {
                  sum+=(checkarr2[i])*base;
                  base*=10;
            }

            if (sum==bineq)
            {
            printf("CORRECT ANSWER    WELL DONE");
            wins=wins+1;
        }
        else
        {
            printf("WRONG ANSWER     CORRECT ANSWER IS:");
            for(int i=0;i<x;i++)
            printf("%d",checkarr2[i]);
            }
    }
```

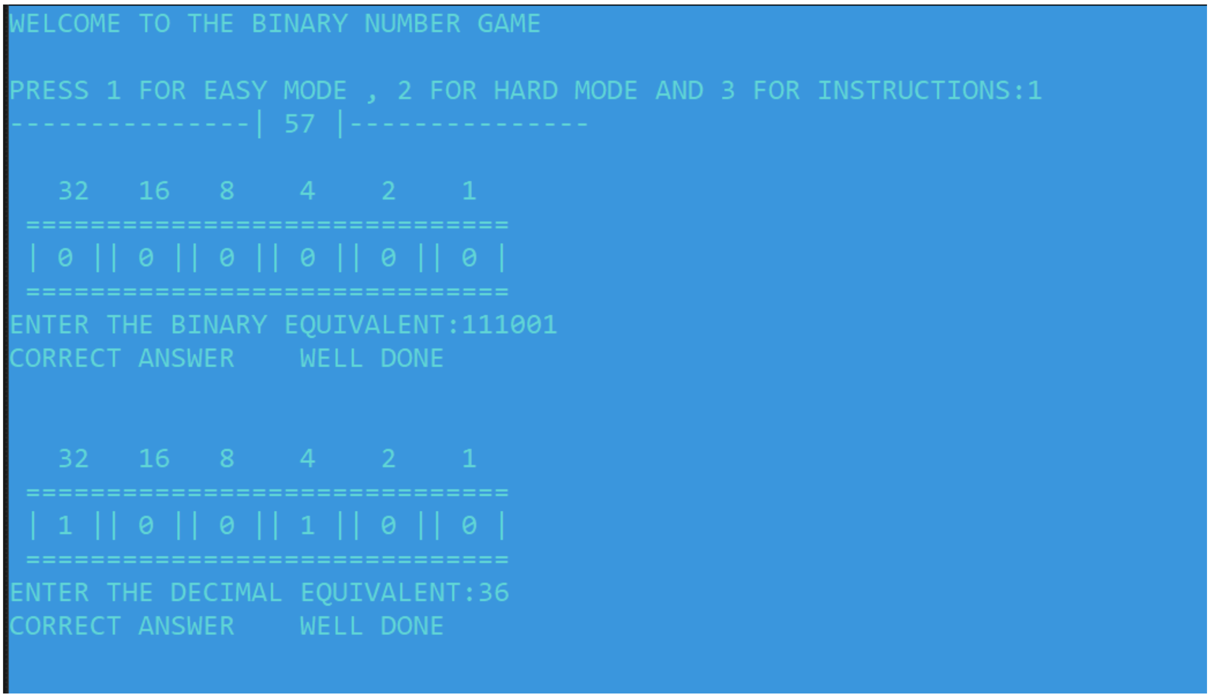
SAMPLE OUTPUT

MENU:-

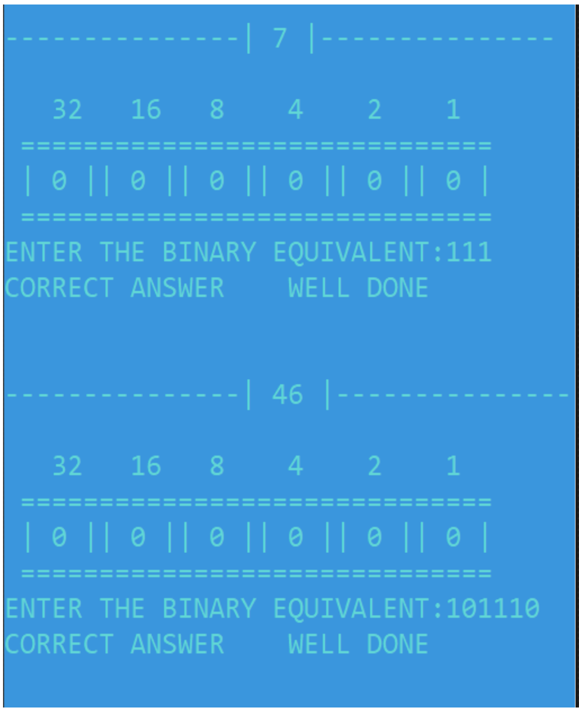


```
WELCOME TO THE BINARY NUMBER GAME

PRESS 1 FOR EASY MODE , 2 FOR HARD MODE AND 3 FOR INSTRUCTIONS:3
FOR EASY MODE:-
For binary to decimal- just add the numbers corresponding to 1 written above
to calculate the decimal equivalent.
For decimal to binary- find the number just lower than the decimal number and
 write 1 in its place. Now subtract this from the original number and repeat
the steps. Insert 0's in between.

FOR HARD MODE:-
 Now imagine the numbers to be written above as in easy mode and proceed in t
he same way.
PRESS 1 FOR EASY MODE , 2 FOR HARD MODE AND 3 FOR INSTRUCTIONS:_
```

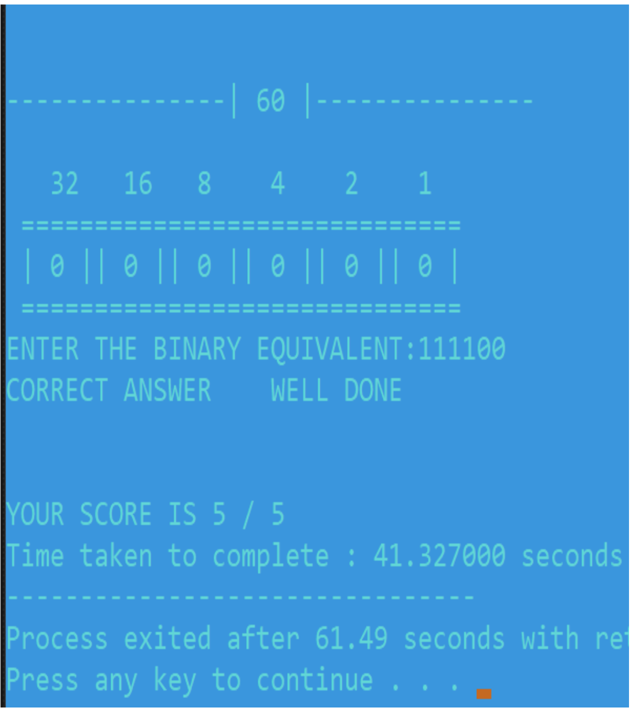
EASY MODE:-



```
WELCOME TO THE BINARY NUMBER GAME

PRESS 1 FOR EASY MODE , 2 FOR HARD MODE AND 3 FOR INSTRUCTIONS:1
---------------| 57 |---------------

   32   16   8    4    2    1
 ==============================
 | 0 || 0 || 0 || 0 || 0 || 0 |
 ==============================
ENTER THE BINARY EQUIVALENT:111001
CORRECT ANSWER    WELL DONE


   32   16   8    4    2    1
 ==============================
 | 1 || 0 || 0 || 1 || 0 || 0 |
 ==============================
ENTER THE DECIMAL EQUIVALENT:36
CORRECT ANSWER    WELL DONE
```

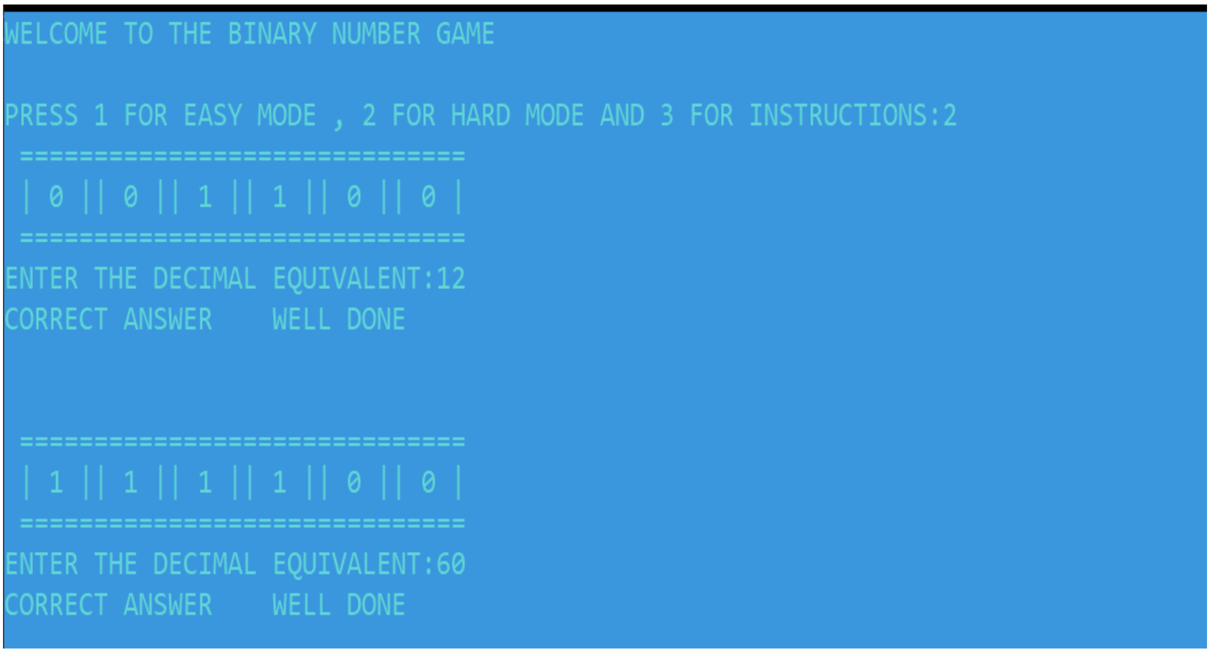


```
---------------| 7 |---------------

   32   16   8    4    2    1
 ===============================
 | 0 || 0 || 0 || 0 || 0 || 0 |
 ===============================
ENTER THE BINARY EQUIVALENT:111
CORRECT ANSWER    WELL DONE


---------------| 46 |---------------

   32   16   8    4    2    1
 ===============================
 | 0 || 0 || 0 || 0 || 0 || 0 |
 ===============================
ENTER THE BINARY EQUIVALENT:101110
CORRECT ANSWER    WELL DONE
```

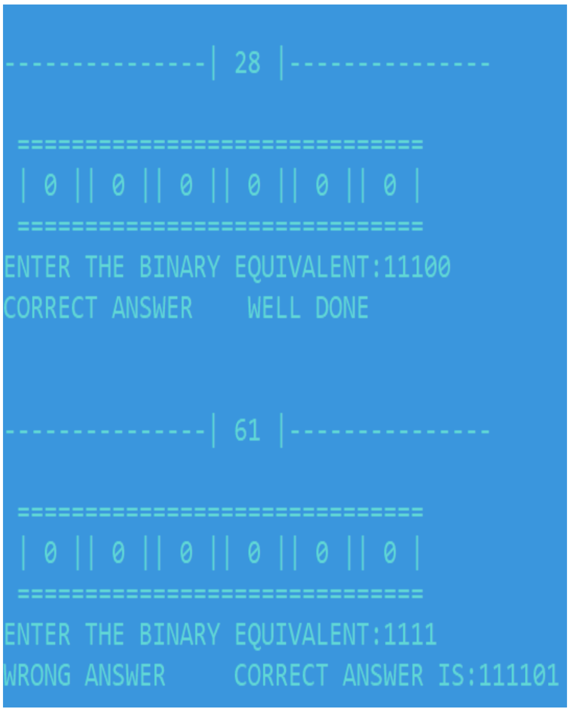


```
---------------| 60 |---------------


   32   16   8    4    2    1
 ===============================
 | 0 || 0 || 0 || 0 || 0 || 0 |
 ===============================
ENTER THE BINARY EQUIVALENT:111100
CORRECT ANSWER    WELL DONE


YOUR SCORE IS 5 / 5
Time taken to complete : 41.327000 seconds
--------------------------------
Process exited after 61.49 seconds with ret
Press any key to continue . . .
```

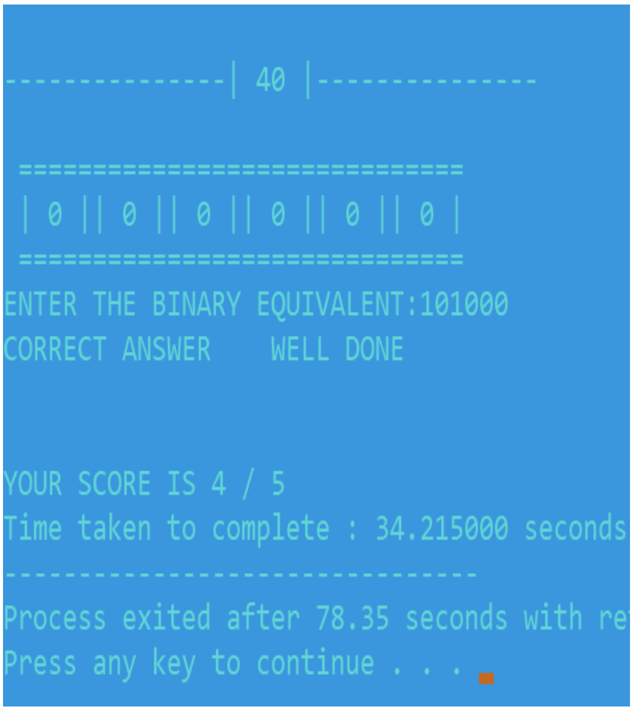
HARD MODE:-



```
WELCOME TO THE BINARY NUMBER GAME

PRESS 1 FOR EASY MODE , 2 FOR HARD MODE AND 3 FOR INSTRUCTIONS:2
 ==============================
 | 0 || 0 || 1 || 1 || 0 || 0 |
 ==============================
ENTER THE DECIMAL EQUIVALENT:12
CORRECT ANSWER    WELL DONE



 ==============================
 | 1 || 1 || 1 || 1 || 0 || 0 |
 ==============================
ENTER THE DECIMAL EQUIVALENT:60
CORRECT ANSWER    WELL DONE
```



```
---------------| 28 |---------------


 ================================
 | 0 || 0 || 0 || 0 || 0 || 0 |
 ================================
ENTER THE BINARY EQUIVALENT:11100
CORRECT ANSWER    WELL DONE



---------------| 61 |---------------


 ================================
 | 0 || 0 || 0 || 0 || 0 || 0 |
 ================================
ENTER THE BINARY EQUIVALENT:1111
WRONG ANSWER     CORRECT ANSWER IS:111101
```



```
---------------| 40 |---------------


 ================================
 | 0 || 0 || 0 || 0 || 0 || 0 |
 ================================
ENTER THE BINARY EQUIVALENT:101000
CORRECT ANSWER    WELL DONE



YOUR SCORE IS 4 / 5
Time taken to complete : 34.215000 seconds
--------------------------------
Process exited after 78.35 seconds with re
Press any key to continue . . .
```

REFERENCES

- For rand () function:- stackoverflow.com

- For clock() function:- geeksforgeeks.com