

211091-Udeesha Prabhashana Jayaweera.

*) Write a pseudo code algorithm and c program to generate binary numbers between 0 to n using a queue.

c code

```
#include <stdio.h>
#include <string.h>
#define MAX 20
char queue[MAX][MAX], temp[MAX];
int front = -1, rear = -1;
void enqueue(char *ptr)
{
    if(rear == MAX-1)
    {
        return;
    }
    if(front == -1 && rear == -1)
        front = rear = 0;
    else
        rear = rear + 1;
    strcpy(queue[rear],ptr);
}
char* dequeue()
{
    if(front == -1)
        printf("\nEmpty Queue");
    else
    {
        strcpy(temp,queue[front]);
        if(front == rear)
            front = rear = -1;
        else
            front = front + 1;
        return temp;
    }
}
void binary_numbers_using_queue()
{
    char temp2[MAX];
    strcpy(temp,dequeue());
    printf("%s\n",temp);
}
```

```

strcpy(temp2,temp);
strcat(temp,"0");
enqueue(temp);
strcat(temp2,"1");
enqueue(temp2);
}
int main()
{
int i,n;
printf("\nEnter the end value : ");
scanf("%d",&n);
char temp[2] = "1";
enqueue(temp);
printf("\nBinary numbers from 1 to %d : \n\n",n);
for(i = 1; i <= n; i++)
binary_numbers_using_queue();
return 0;
}

```

pseudo code

begin

Input the end_value up to which the binary numbers have to be generated.

Enqueue "1" to the queue.

Initialize string a = queue.front()

Dequeue a value from the queue.

Print a.

Initialize string b = a.

Append "0" to string a and enqueue it to the stack.

Append "1" to string b and enqueue it to the stack.

Repeat steps 3 to 8 until the end value.

end