Design and Analysis of Algorithms Assignment - 4

Name: Dhanraj Kore

Div: TY B

Roll No: 60

Batch : B-3

Josephus Problem

Approach 1: *Using Circular Linked List*

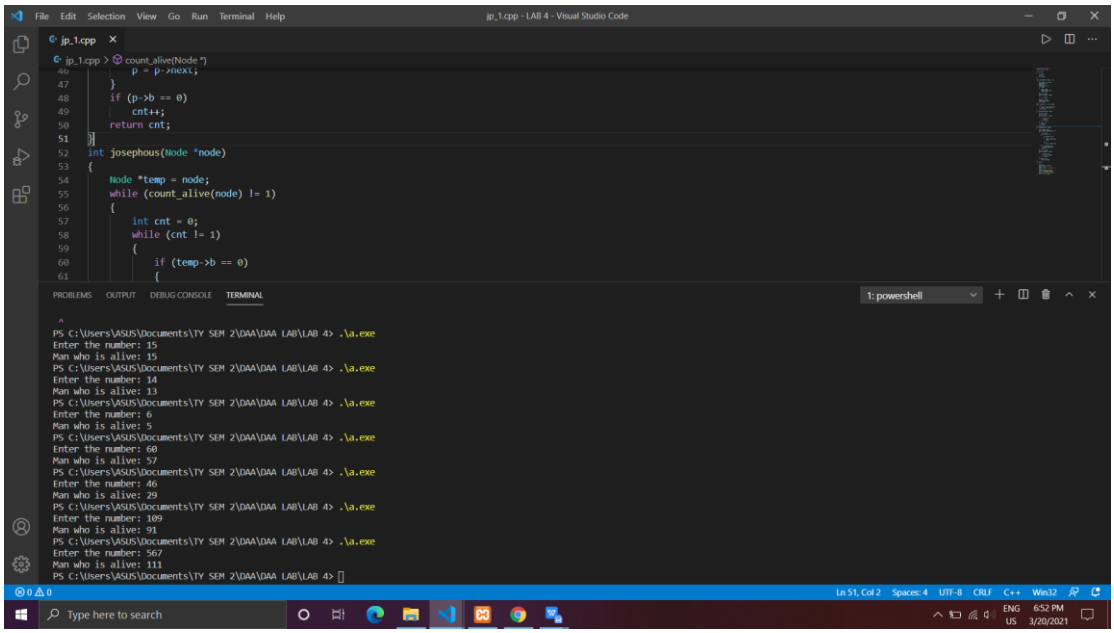
CODE :

```
#include <bits/stdc++.h>
using namespace std;


struct Node
{
    int data;
    int b;
    Node *next;
};
Node *insert(Node *node, int x)
{
    Node *temp = new Node;
    temp->data = x;
    temp->b = 0;
    temp->next = NULL;
    if (!node)
    {
        node = temp;
        temp->next = node;
        return node;
    }
```

```
    Node *p = node;
    while (p->next != node)
    {
        p = p->next;
    }
    p->next = temp;
    temp->next = node;
    return node;
}
Node *create(int n, Node *node)
{
    for (int i = 1; i <= n; i++)
        node = insert(node, i);
    return node;
}
int count_alive(Node *node)
{
    Node *p = node;
    int cnt = 0;
    while (p->next != node)
    {
        if (p->b == 0)
            cnt++;
        p = p->next;
    }
    if (p->b == 0)
        cnt++;
    return cnt;
}
int josephous(Node *node)
{
    Node *temp = node;
    while (count_alive(node) != 1)
    {
        int cnt = 0;
        while (cnt != 1)
```

```
        {
            if (temp->b == 0)
            {
                temp = temp->next;
                cnt++;
            }
            else
                temp = temp->next;
        }
        while (temp->b == 1)
            temp = temp->next;
        temp->b = 1;
    }
    Node *p = node;
    while (p->next != node)
    {
        if (p->b == 0)
            return p->data;
        p = p->next;
    }
    if (p->b == 0)
        return p->data;
}
int main()
{
    int n;
    Node *node = new Node;
    node = NULL;
    cout << "Enter the number: ";
    cin >> n;
    node = create(n, node);
    int ans = josephous(node);
    cout << "Man who is alive: ";
    cout << ans << endl;
}
```

O/P :



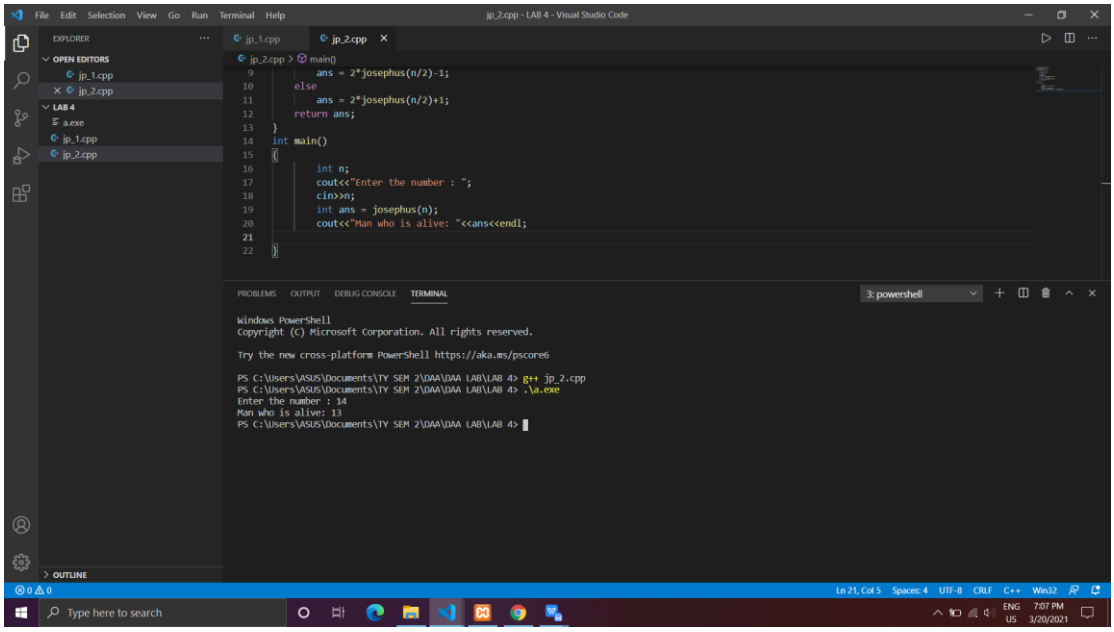
Approach 2 : *Using Recursion*

CODE :

```
#include<bits/stdc++.h>

using namespace std;

int josephus(int n)

{

    if(n==0 || n==1)

        return n;

    int ans;

    if(n%2==0)

        ans = 2*josephus(n/2)-1;

    else

        ans = 2*josephus(n/2)+1;

    return ans;

}
```

```
int main()

{

        int n;

        cout<<"Enter the number : ";

        cin>>n;

        int ans = josephus(n);

        cout<<"Man who is alive: "<<ans<<endl;


}
```

O/P :



Approach 2 : *Using Bit Manipulation*

CODE :

```
#include <bits/stdc++.h>

using namespace std;


int main()
```

```cpp
{

    int n;

    cout << "Enter the number : ";

    cin >> n;

    int m = 1, cnt = 1;

    ;

    while (m < n)

        m = pow(2, cnt++);

    int ans;

    if (n == pow(2, cnt - 1))

        ans = 2 * (n - m) + 1;

    else

    {

        m /= 2;

        ans = 2 * (n - m) + 1;

    }

    cout << "Man who is alive: " << ans << endl;

}
```

O/P :