**NAME:ROHAN NYATI**

**SAP ID:500075940**

**ROLL NO. : R177219148**

**BATCH-5(AI&ML)**

**Josephus Problem -**

There are n men standing in a circle. Each man will kill the man next to him is clockwise order starting from the man at position 1 .Find the position of the man that survives.

If n is a power of 2 say n = 2^a. Then 1 will always survive as after each cycle, exactly half the men remain and each cycle ends at 1. Infact, whichever position we begin with will survive the purge.

Any number n can be represented as 2^a + l where 2^a is the largest power of 2 smaller than n.

After eliminating l people, we are at position 2\*l + 1. Now there are only 2^a people left which is a power of 2. So the person at 2\*l + 1 will survive, since the remaining number is a power of 2.

--- IMPLEMENTATION IN PYTHON3 ---

import math

def josephus\_solve( n ) :

""" The solution to the josephus problem is ---->

2 \* l + 1 where n = 2^a + l where 2^a is the

largest power of 2 smaller than n """

a = math.log( n , 2 )

a = math.floor( a )

l = n - 2\*\*a

return 2\*l + 1

if \_\_name\_\_ == "\_\_main\_\_" :

n = int( input("Enter the number of seats : ") )

winning\_seat = josephus\_solve( n )

print( "The winning seat for josephus is --> " , winning\_seat )

--- OUTPUT ---

