SANYAM AGRAWAL – SE21UCSE192 – CSE3

DAA Lab 7

Island.c

```c
#include <stdio.h>
#include <stdlib.h>
#include <time.h>
double calc_prob(int N, int x, int y, int n)
{
    // If person is outside matrix, they are not alive.
    if (x < 0 || x >= N || y < 0 || y >= N)
    {
        return 0.0;
    }
    // If no steps left to take, the person is alive at this location.
    if (n == 0)
    {
        return 1.0;
    }
    // Initialize the prob to 0.0
    double prob = 0.0;
    // Move in all four directions, and recursively calculate the prob
    prob += 0.25 * calc_prob(N, x + 1, y, n - 1); // Move in east direction
    prob += 0.25 * calc_prob(N, x - 1, y, n - 1); // Move in west direction
    prob += 0.25 * calc_prob(N, x, y + 1, n - 1); // Move in south direction
    prob += 0.25 * calc_prob(N, x, y - 1, n - 1); // Move in north direction
    return prob;
}
int main()
{
    int N;    // Size of the island
    int x, y; // Starting position of person
    int n;    // Number of steps taken by person
```

Island.c

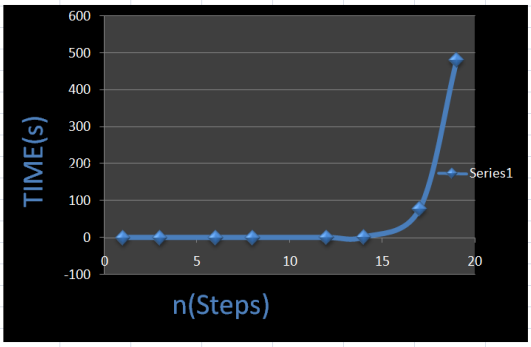
Island.c > main()

```c
    printf("Enter the size of island (N): ");
    scanf("%d", &N);
    printf("Enter starting position of person (x, y): ");
    scanf("%d %d", &x, &y);
    printf("Enter the number of steps (n): ");
    scanf("%d", &n);

        // If starting Location is invalid (out of index)
    if (x < 0 || x >= N || y < 0 || y >= N)
    {
        printf("Invalid Starting Location");
    }
    else
    {
        clock_t start_time = clock();
        // Calculate the prob
        double prob = calc_prob(N, x, y, n);

        clock_t end_time = clock();
        double elapsed_time = (double)(end_time - start_time) / CLOCKS_PER_SEC;
        printf("Size: %d\tTime: %f\n", n, elapsed_time);

        // Output the result
        printf("Probability that the person is alive after %d steps is: %lf\n", n, prob);
    }
    return 0;
}
```

Complexity Analysis:

The time it takes to run the program depends on how many steps the person takes and where they start on the island. If the person takes more steps, the program takes longer. Also, if the person starts closer to the edge of the island, it's faster than if they start in the middle. The size of the island matters too - bigger islands make the program slower. Here, I fixed sixe of island to N(20) and change the n(steps) and position(x,y) of person.

n(Steps)	TIME
1	0.000003
3	0.000004
6	0.000045
8	0.000677
12	0.094078
14	1.340524
17	77.49093
19	480.5748



-> The time it takes to run the program depends on how many steps the person takes and where they start on the island. If the person takes more steps, the program takes longer. Also, if the person starts closer to the edge of the island, it's faster than if they start in the middle. The size of the island matters too - bigger islands make the program slower. Here, I fixed sixe of island to N(20) and change the n(steps) and position(x,y) of person.

Sanyam Agrawal - SE21UCSE192 - CSE3