Activity 2

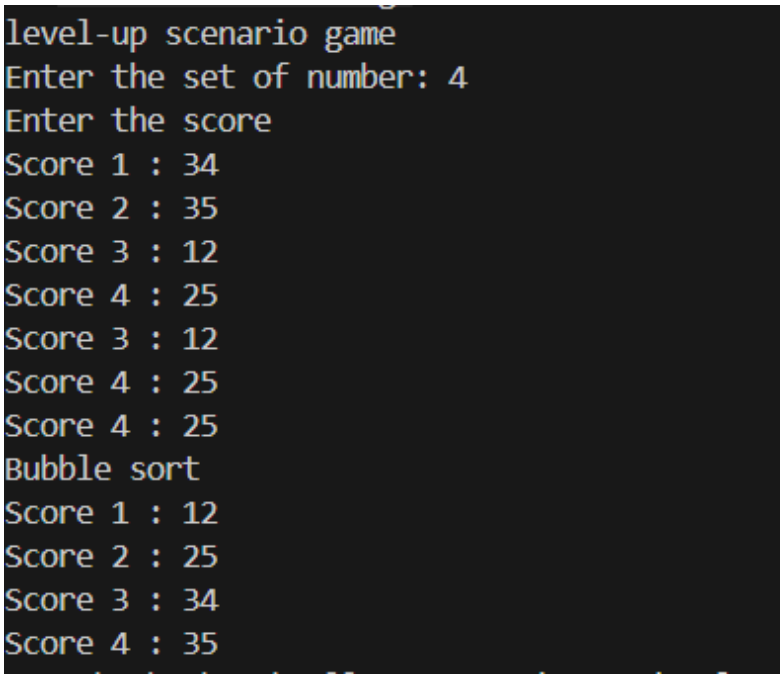
Game Development:

write a game development program that implements the Bubble Sort algorithm. The program will simulate a simple game where the player can input a set of numbers, and the numbers will be sorted using Bubble Sort to simulate a "level-up" scenario where the player's scores are sorted in ascending order.

```
#include <iostream>
using namespace std;

void bubblesort(int score[], int size){
	for(int i=0; i<size-1; i++){
		for(int j=0; j<size-i-1; j++){
			if(score[j] > score[j+1]){
				int temp = score[j];
				score[j] = score[j+1];
				score[j+1] = temp;
				}
			}
		}

	}

int main(){
	int size;

	cout<<"level-up scenario game"<<endl;
	cout<<"Enter the set of number: ";
	cin>>size;

	int score[size];

	cout<<"Enter the score"<<endl;
	for(int i=0; i<size; i++){
		cout<<"Score "<<i+1<<" : ";
		cin>>score[i];
		}
```

```
bubblesort(score, size);

cout<<"Bubble sort "<<endl;

for(int i=0; i<size; i++){
        cout<<"Score "<<i+1<<" : "<<score[i]<<endl;
        }



return 0;
}
```

Output:



Insertion Sort:

Organizing Cards in a Hand:

Application: When playing card games, players often use an approach similar to insertion sort to organize their cards. They pick one card at a time and insert it into the correct position in their hand, maintaining a sorted sequence. Write a program that demonstrates how to organize (sort) cards in a hand using insertion sort

```
#include <iostream>
using namespace std;

void displayHand(int cards[], int n) {
    for (int i = 0; i < n; i++) {
        cout << cards[i] << " ";
    }
    cout << endl;
}

int main() {
    int n;
    cout << "Enter the number of cards: ";
    cin >> n;

    int cards[n];
    for (int i = 0; i < n; i++) {
        cout << "Card no " << i + 1 << " is: ";
        cin >> cards[i];
    }

    cout << "\nSorting process using Insertion Sort:\n";
    for (int i = 1; i < n; i++) {
        int key = cards[i];
        int j = i - 1;
        while (j >= 0 && cards[j] > key) {
            cards[j + 1] = cards[j];
            j--;
        }
        cards[j + 1] = key;
        cout << "After inserting card " << i + 1 << ": ";
        displayHand(cards, i + 1);
    }
```

```
    cout << "\nFinal sorted hand: ";
    displayHand(cards, n);

    return 0;
}
```

```
Enter the number of cards: 8
Enter the 8 cards: 23
45
78
90
12
1
23
4
After the sorting of cards.
1
4
12
23
23
45
78
90


------------------
(program exited with code: 0)

Press any key to continue . . .
```