1 Correctness of Selection Sort

Selection sort is a sorting algorithm that finds the smallest number in the list and inserts it at the beginning of the list while shifting the "beginning" up every loop.

```
def selection_sort(arr):
    array_length = len(arr)

    for i in range(array_length - 1):
        min_index = i  # Assume the first elemenpress Nt is the minimum

        # Find the minimum element in the remaining unsorted array
        for j in range(i + 1, array_length):
            if arr[j] < arr[min_index]:
                min_index = j

        # Swap the found minimum element with the first element of the unsorted part
        arr[i], arr[min_index] = arr[min_index], arr[i]

    return arr
```

2 Loop Invariant

Initial: At the start, the subarray $A[0:i]$ is sorted.

Maintenance: Assume that at the start of iteration i, the subarray $A[0:i]$ is sorted. The inner loop finds the smallest element in $A[i:n]$ and places it at $A[i]$ so it remains sorted.

Termination: all elements have been placed in their correct positions, so $A[0:n]$ is sorted.

3 Conclusion

The loop invariant holds at every iteration and by termination, the entire array is sorted.

Thus, Selection Sort correctly sorts an array of size n.