Correctness of selection sort:

- Algorithm → find the smallest element in the unsorted part of the array and swap it with the first element in the unsorted part of the array.

```
for i = 0 to n-1 {
    minimum = i
    for j = i+1 to n {
        if (arr [j] < arr [min]) {
            minimum = j
        }
    }
    if ( i != minimum) {
        swap ( ith element, minimum element))
    }
}.
```

- correctness -

i) Initialization:-

```
for i=0 to n-1 {
        minimum = i
```

We ~~ran~~ start the outer loop by assuming index of smallest element ~~for~~ in sub array i to n-1 is at index i.

ii) Maintaiance:-

```
for j=i+1 to n
    if   arr [j] < a[minimum]
            minimum=j.
swap (arr [i], arr [minimum])
```

- before the execution of inner loop, all the elements before i are sorted.
- The inner loop identifies the minimum element in the ~~array~~ sub array [i to n] and updates minimum value accordinly.
- The algorithm sorts the element at index i which element at index minimum.

iii) Termination! The outer loop runs from 0 to n-1, and it iterates through entire array. The algorithm terminates when the outer loop completes, There are no infinite loops or conditions the prevent the algo from completing the sorting proces