



1. Disprove the claim: "For all integers x , $x^2 - 3x + 2 \geq 0$ ".
2. Find out the loop invariant conditions in the following selection sort algorithm.

```
min_idx = 0

for (i = 0; i < n-1; i++)
{
    min_idx = i;
    for (j = i+1 to n-1)
        if (arr[j] < arr[min_idx])
            min_idx = j;

    swap(&arr[min_idx], &arr[i]);
}
```

3. Prove that the sum of the first n powers of 2 is $2^n - 1$ is correct.
4. Find out the loop invariant conditions in the following insertion sort algorithm.

```
for (i = 1 to n-1)
{
    key = arr[i];
    j = i-1;
    while (j >= 0 and arr[j] > key)
    {
        arr[j+1] = arr[j];
        j = j-1;
    }
    arr[j+1] = key;
}
```

5. Prove the correctness of the insertion sort algorithm using loop invariant.

```
algorithm insertionSort(a):
    // INPUT
    //   a = an array of n real numbers (1-based indexing)
    // OUTPUT
    //   The non-decreasingly ordered permutation of a

    for j ← 2 to n:
        x ← a[j]
        i ← j - 1
        while i > 0 and a[i] > x:
            a[i + 1] ← a[i]
            i ← i - 1
        a[i + 1] ← x
    return a
```

6. Prove the correctness of the following algorithm to compute the sum of elements in an array:

```
1  #include <stdio.h>
2
3  int main() {
4      int n;
5      printf("Enter the number of elements in the array: ");
6      scanf("%d", &n);
7
8      int A[n], sum = 0; // Initialize the array and the sum
9      printf("Enter the elements of the array:\n");
10     for (int i = 0; i < n; i++) {
11         scanf("%d", &A[i]); // Input each element
12     }
13     // Compute the sum directly in the main function
14     for (int i = 0; i < n; i++) {
15         sum += A[i]; // Add each element to the sum
16     }
17
18     printf("The sum of the array elements is: %d\n", sum);
19     return 0;
20 }
```