



Lab 5

Exercises

1. (0.5p) Find the right specifications for the method that returns both the minimum value AND its index below:

```
method ArrayMinIndex(a: array<int>)
    returns (min: int, minIndex: int)
    // Add your requires clause here
    // Add your ensures clauses here
{
    min := a[0];
    minIndex := 0;
    var i := 1;

    while i < a.Length
        // Add your invariants here
    {
        if a[i] > min {
            min := a[i];
            minIndex := i;
        }
        i := i + 1;
    }
}
```

2. (0.5p) Write a method that returns the difference between maximum and minimum elements of an array. Write the complete specifications of this program and verify them using Dafny.

3. (0.5p) Specify the postconditions and loop invariants for linear search in an array.

```
method Find(a: array<int>, key: int) returns (index: int)
    // Add your ensures clauses here
{
    index := 0;

    while index < a.Length
        // Add your invariants here
    {
        if a[index] == key {
            return;
        }
        index := index + 1;
    }

    index := -1;
}
```

4. (0.5p) Write a method that counts how many times a value appears in an array. Write the complete specifications of this program and verify them using Dafny.

5. (1p) Write a method that finds the LAST occurrence of a value in an array. Write the complete specifications of this program and verify them using Dafny.

6. (1p) Add all specifications for binary search on a sorted array.

```
predicate sorted(a: array<int>)
    reads a
{
    forall j, k :: 
        0 <= j < k < a.Length ==> a[j] <= a[k]
}

method BinarySearch(a: array<int>, key: int)
    returns (index: int)
    // Add your requires clause here
    // Add your ensures clauses here
{
    var low := 0;
    var high := a.Length;

    while low < high
        // Add your invariants here
    {
        var mid := (low + high) / 2;

        if a[mid] < key {
            low := mid + 1;
        } else if a[mid] > key {
            high := mid;
        } else {
            return mid;
        }
    }

    return -1;
}
```

BONUS

7. (1p) Modify binary search to find the FIRST occurrence in a sorted array with duplicates. Write the complete specifications of this program and verify them using Dafny.

Last updated: 11/06/2025 09:13:26 - Copyright © Andrei Arusoaei

Facultatea de Informatică, Universitatea „Alexandru Ioan Cuza” din Iași