

Automatic and Interactive Reasoning

Exercise Sheet, Week 4

Ștefan Ciobâcă

20.10.2025

Exercise Sheet

Paper Exercises (please do not use AI to solve these exercises)

Consider the following Dafny code implementing binary search:

```
method {:isolate_assertions} binarySearch(s : seq<int>, key : int) returns (r : int)
  requires forall j, k :: 0 <= j < k < |s| ==> s[j] <= s[k]
  ensures r >= 0 ==> 0 <= r < |s| && s[r] == key
  ensures r < 0 ==> forall k :: 0 <= k < |s| - 1 ==> s[k] != key
{
  var left : int := 0;
  var right : int := |s| - 1;
  while (left <= right)
    invariant 0 <= left <= |s|
    invariant -1 <= right < |s|
    invariant forall k :: 0 <= k < left ==> s[k] < key
    invariant forall k :: right < k < |s| ==> s[k] > key
    decreases right - left
  {
    var mid : int := (left + right) / 2;
    if (key < s[mid]) {
      right := mid - 1;
    } else if (key > s[mid]) {
      left := mid + 1;
    } else {
      return mid;
    }
  }
}
return -1;
}
```

Write down (using pen and paper) what are the verification conditions for the following facts:

1. the invariants hold on entry;
2. the invariants are maintained by the loop body;
3. the postcondition holds.

Application (you may use AI)

1. Install Dafny (<https://github.com/dafny-lang/>).
2. Install Boogie (<https://github.com/boogie-org/boogie>).
3. Generate the SMT-LIB verification conditions generated by the Dafny code above using the following commands:

```
dotnet Dafny.dll -print:binsearch.bpl binsearch.dfy
```

```
dotnet BoogieDriver.dll /timeLimit:5 /trace /proverLog:binsearch.smt2 binsearch.bpl
```

(Save the `binsearch.smt2` file for later)

4. Identify in the `binsearch.smt2` file the verification obligations for the algorithm (which assertion in the SMT-LIB file corresponds to which verification obligation).