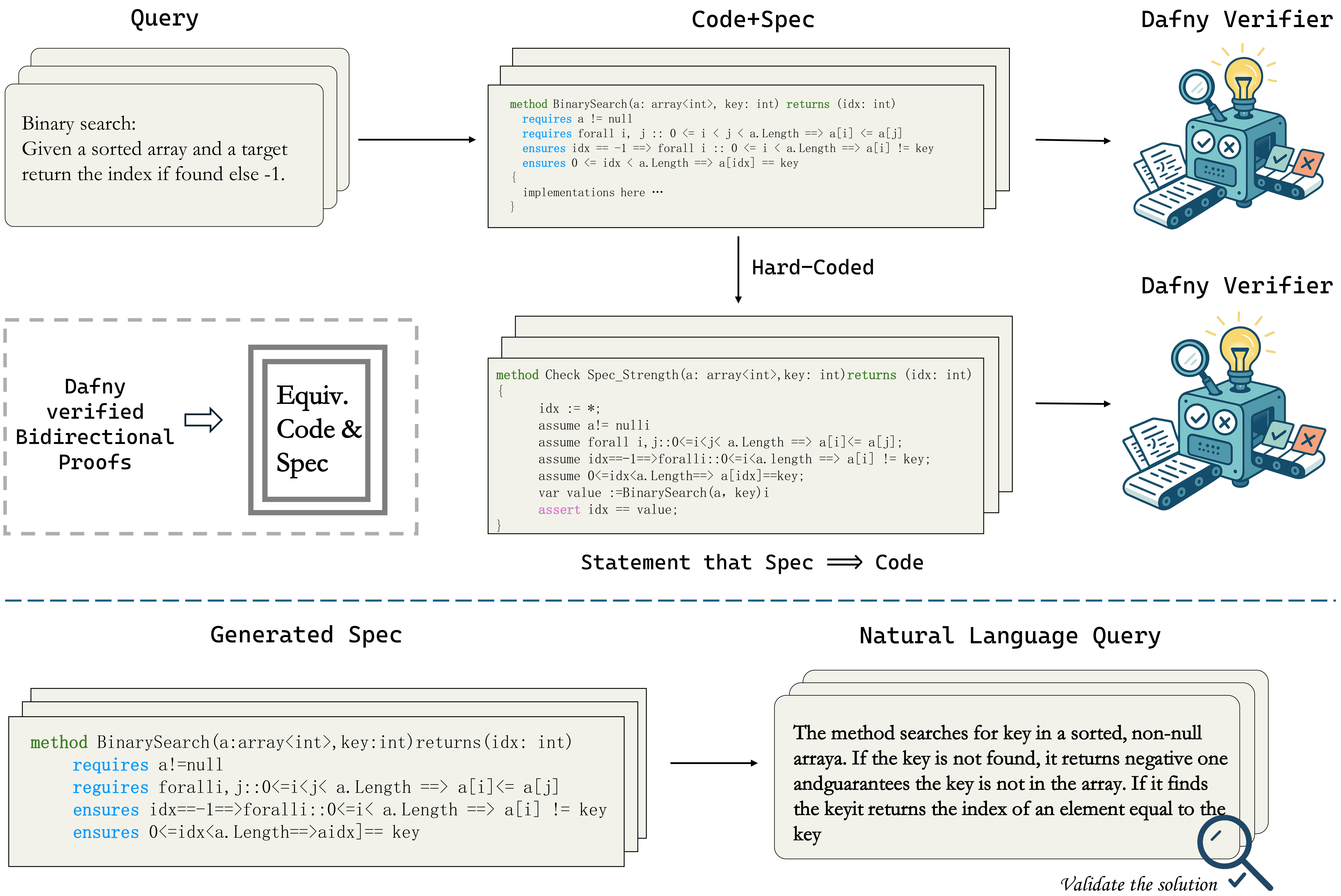

Binary search:
Given a sorted array and a target
return the index if found else -1.

```
method BinarySearch(a: array<int>, key: int) returns (idx: int)
  requires a != null
  requires forall i, j :: 0 <= i < j < a.Length ==> a[i] <= a[j]
  ensures idx == -1 ==> forall i :: 0 <= i < a.Length ==> a[i] != key
  ensures 0 <= idx < a.Length ==> a[idx] == key
{
  implementations here ···
}
```

```
method Check Spec_Strength(a: array<int>,key: int)returns (idx: int)
{
      idx := *;
      assume a!= nulli
      assume forall i,j::0<=i<j< a.Length ==> a[i]<= a[j];
      assume idx==-1==>foralli::0<=i<a.length ==> a[i] != key;
      assume 0<=idx<a.Length==> a[idx]==key;
      var value :=BinarySearch(a, key)i
      assert idx == value;
}
```

```
method BinarySearch(a:array<int>,key:int)returns(idx: int)
    requires a!=null
    reguires foralli,j::0<=i<j< a.Length ==> a[i]<= a[j]
    ensures idx==-1==>foralli::0<=i< a.Length ==> a[i] != key
    ensures 0<=idx<a.Length==>aidx]== key
```

The method searches for key in a sorted, non-null arraya. If the key is not found, it returns negative one andguarantees the key is not in the array. If it finds the keyit returns the index of an element equal to the key