

Experiment ‘p4_4_month_with_highest_temperature’ Results

December 18, 2025

Experiment outcome: SUCCESS

Bad responses: 0

Responses containing assume : 0

Resolution attempts: 1

Hard fails (resolution): 0

Soft fails (resolution): 0

Verification attempts: 1

Problem Specification

Problem name: p4_4_month_with_highest_temperature

Natural language statement: Write a method that reads a sequence of twelve temperature values and returns the number of the month with the highest temperature.

Method signature: p4_5_month_with_highest_temperature(inputs: seq<real>) returns (hottest: int)

Ensures

- $1 \leq \text{hottest} \leq 12$
- $\forall i :: 0 \leq i \leq |\text{inputs}| - 1 \implies \text{inputs}[\text{hottest}-1] \geq \text{inputs}[i]$

Requires

- $|\text{inputs}| == 12$

GenAI interactions

Below you will find all interactions between the ‘user’ (program) and the ‘assistant’ (OpenAI).

Program → GenAI

You are given the following task to perform in Dafny:

Write a method that reads a sequence of twelve temperature values and returns the number ↪ of the month with the highest temperature.

The signature should be:

```
method p4_5_month_with_highest_temperature(inputs: seq<real>) returns (hottest: int)
```

The method should respect the following contract:

```
requires |inputs| == 12, ensures 1 <= hottest <= 12, ensures forall i :: 0 <= i <= |  
↪ inputs|-1 ==> inputs[hottest-1] >= inputs[i]
```

Produce and show only the Dafny body of this method, including the curly braces that
↪ surround it. Do not show the signature nor contract. You must not use ‘assume’
↪ anywhere in your code.

GenAI → Program

System fingerprint:

ID: chatcmpl-Co4e2byscvOVSQchlY9CwOLkyJnj9

Model: gpt-5.1-chat-latest

Created at: 1766050290

Finish reason: stop

Usage: 342 tokens in, and 158 tokens out

```
{  
    var maxIndex := 0;  
    var maxVal := inputs[0];  
    var i := 1;  
    while i < 12  
        invariant 1 <= i <= 12  
        invariant 0 <= maxIndex < i  
        invariant maxVal == inputs[maxIndex]  
        invariant forall j :: 0 <= j < i ==> maxVal >= inputs[j]  
    {  
        if inputs[i] > maxVal {  
            maxVal := inputs[i];  
            maxIndex := i;  
        }  
        i := i + 1;  
    }  
    hottest := maxIndex + 1;  
}
```

Final Program

```
method p4_5_month_with_highest_temperature(inputs: seq<real>) returns (hottest: int)
    requires |inputs| == 12
    ensures 1 <= hottest <= 12
    ensures forall i :: 0 <= i <= |inputs|-1 ==> inputs[hottest-1] >= inputs[i]
{
    var maxIndex := 0;
    var maxVal := inputs[0];
    var i := 1;
    while i < 12
        invariant 1 <= i <= 12
        invariant 0 <= maxIndex < i
        invariant maxVal == inputs[maxIndex]
        invariant forall j :: 0 <= j < i ==> maxVal >= inputs[j]
    {
        if inputs[i] > maxVal {
            maxVal := inputs[i];
            maxIndex := i;
        }
        i := i + 1;
    }
    hottest := maxIndex + 1;
}
```

Total Token Usage

Input tokens: 342
Output tokens: 158
Reasoning tokens: 0
Sum of ‘total tokens’: 500

Experiment Timings

Overall Experiment started at 1766050291926, ended at 1766050295548, lasting 3622ms (3.62 seconds)
Iteration #1 started at 1766050291926, ended at 1766050295548, lasting 3622ms (3.62 seconds)