

# Experiment ‘P\_3\_5\_Order’ 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:** P\_3\_5\_Order

**Natural language statement:** Write a method that takes three numbers and returns "increasing" if they are in increasing order, "decreasing" if they are in decreasing order, and "neither" otherwise. Here, "increasing" means "strictly increasing", with each value larger than its predecessor. The sequence 3 4 4 would not be considered increasing.

**Method signature:** P\_3\_5\_Order (i:int, j: int, k: int) returns (s: string)

### Ensures

- if i < j < k then s == "increasing" else if i > j && j > k then s == "decreasing" else s== "neither"

## 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 takes three numbers and returns "increasing" if they are in  
→ increasing order, "decreasing" if they are in decreasing order, and "neither"  
→ otherwise. Here, "increasing" means "strictly increasing", with each value larger  
→ than its predecessor. The sequence 3 4 4 would not be considered increasing.

The signature should be:

```
method P_3_5_Order (i:int, j: int, k: int) returns (s: string)
```

The method should respect the following contract:

```
ensures if i < j < k then s == "increasing" else if i > j && j > k then s == "decreasing"  
→ else s== "neither"
```

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: chatmpl-Co4Se0sHq29NHzjjJL3SUYHURJEG4

Model: gpt-5.1-chat-latest

Created at: 1766049584

Finish reason: stop

Usage: 438 tokens in, and 143 tokens out

```
{  
    if i < j && j < k {  
        s := "increasing";  
    } else if i > j && j > k {  
        s := "decreasing";  
    } else {  
        s := "neither";  
    }  
}
```

### Final Program

```
method P_3_5_Order (i:int, j: int, k: int) returns (s: string)  
    ensures if i < j < k then s == "increasing" else if i > j && j > k then s == "  
→ decreasing" else s== "neither"
```

```
{  
    if i < j && j < k {  
        s := "increasing";  
    } else if i > j && j > k {  
        s := "decreasing";  
    } else {  
        s := "neither";  
    }  
}
```

## Total Token Usage

**Input tokens:** 438

**Output tokens:** 143

**Reasoning tokens:** 64

**Sum of ‘total tokens’:** 581

## Experiment Timings

**Overall Experiment** started at 1766049586651, ended at 1766049590221, lasting 3570ms (3.57 seconds)

**Iteration #1** started at 1766049586651, ended at 1766049590221, lasting 3570ms (3.57 seconds)