

# Experiment 'p6\_2\_a\_swap\_first\_last' 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:** p6\_2\_a\_swap\_first\_last

**Natural language statement:** Write a method to swap the first and last elements in an array.

**Method signature:** p6\_2\_a\_swap\_first\_last(arr: seq<int>) returns (result: seq<int>)

### Ensures

- `|result| == |arr|`
- `result[0] == arr[|arr| - 1]`
- `result[|arr| - 1] == arr[0]`
- `forall i :: 1 <= i < |arr| - 1 ==> result[i] == arr[i]`

### Requires

- `|arr| >= 2`

## 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 to swap the first and last elements in an array.

The signature should be:

method p6\_2\_a\_swap\_first\_last(arr: seq<int>) returns (result: seq<int>)

The method should respect the following contract:

requires |arr| >= 2, ensures |result| == |arr|, ensures result[0] == arr[|arr| - 1],  
    ↪ ensures result[|arr| - 1] == arr[0], ensures forall i :: 1 <= i < |arr| - 1 ==>  
    ↪ result[i] == arr[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-Co55qq2bYjRooPfeUvzUNiEB6tEAF

**Model:** gpt-5.1-chat-latest

**Created at:** 1766052014

**Finish reason:** stop

**Usage:** 380 tokens in, and 181 tokens out

```
{  
  result := [arr[|arr| - 1]] + arr[1..|arr| - 1] + [arr[0]];  
}
```

## Final Program

```
method p6_2_a_swap_first_last(arr: seq<int>) returns (result: seq<int>)  
  requires |arr| >= 2  
  ensures |result| == |arr|  
  ensures result[0] == arr[|arr| - 1]  
  ensures result[|arr| - 1] == arr[0]  
  ensures forall i :: 1 <= i < |arr| - 1 ==> result[i] == arr[i]  
{  
  result := [arr[|arr| - 1]] + arr[1..|arr| - 1] + [arr[0]];  
}
```

## Total Token Usage

Input tokens: 380

Output tokens: 181

Reasoning tokens: 128

Sum of 'total tokens': 561

## Experiment Timings

**Overall Experiment** started at 1766052016129, ended at 1766052019928, lasting 3799ms (3.80 seconds)

**Iteration #1** started at 1766052016129, ended at 1766052019928, lasting 3799ms (3.80 seconds)