

Experiment ‘p6_2_d_replace_with_larger_neighbor’ Results

December 18, 2025

Experiment outcome: SUCCESS

Bad responses: 0

Responses containing assume : 0

Resolution attempts: 6

Hard fails (resolution): 5

Soft fails (resolution): 0

Verification attempts: 1

Problem Specification

Problem name: p6_2_d_replace_with_larger_neighbor

Natural language statement: Write a method to replace each element of an array, except the first and last, by the larger of its two neighbors.

Method signature: p6_2_d_replace_with_larger_neighbor(arr: seq<int>) returns (result: seq<int>)

Ensures

- $|result| == |arr|$
- $result[0] == arr[0]$
- $result[|arr| - 1] == arr[|arr| - 1]$
- $\forall i :: 1 \leq i < |arr| - 1 \implies result[i] == \begin{cases} arr[i - 1] & \text{if } arr[i - 1] \geq arr[i + 1] \\ arr[i + 1] & \text{else} \end{cases}$

Requires

- $|arr| \geq 3$

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 replace each element of an array, except the first and last, by the ↪ larger of its two neighbors.

The signature should be:

```
method p6_2_d_replace_with_larger_neighbor(arr: seq<int>) returns (result: seq<int>)
```

The method should respect the following contract:

```
requires |arr| >= 3, ensures |result| == |arr|, ensures result[0] == arr[0], ensures
    ↪ result[|arr| - 1] == arr[|arr| - 1], ensures forall i :: 1 <= i < |arr| - 1 ==>
    ↪ result[i] == if arr[i - 1] >= arr[i + 1] then arr[i - 1] else arr[i + 1]
```

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-Co56MKd4Tv7Pbgjho8YXik8WkAtJN

Model: gpt-5.1-chat-latest

Created at: 1766052046

Finish reason: stop

Usage: 456 tokens in, and 285 tokens out

```
{
  var mid := seq i | 1 <= i < |arr| - 1 :: if arr[i - 1] >= arr[i + 1] then arr[i - 1]
    ↪ else arr[i + 1];
  result := arr[0..1] + mid + arr[|arr| - 1..];
```

Program → GenAI

When using dafny resolve, the below error is emitted and resolve fails:

```
problem_data/p6_2_d_replace_with_larger_neighbor/program.dfy(8,17): Error: openparen
    ↪ expected
  |
8 | var mid := seq i | 1 <= i < |arr| - 1 :: if arr[i - 1] >= arr[i + 1] then arr[i - 1]
    ↪ else arr[i + 1];
  | ^
```

```

problem_data/p6_2_d_replace_with_larger_neighbor/program.dfy(8,40): Error: comma expected
|
8 | var mid := seq i | 1 <= i < |arr| - 1 :: if arr[i - 1] >= arr[i + 1] then arr[i - 1]
  ↵ else arr[i + 1];
  |
  ^

2 parse errors detected in program.dfy

Correct the error by altering only the method body. Produce and show only the Dafny body,
  ↵ 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-Co56QzzRQ1FGoQvREuWvoQa5WCkRE

Model: gpt-5.1-chat-latest

Created at: 1766052050

Finish reason: stop

Usage: 810 tokens in, and 159 tokens out

```
{
    var mid := seq(i | 1 <= i && i < |arr| - 1 :: if arr[i - 1] >= arr[i + 1] then arr[i -
      ↵ 1] else arr[i + 1]);
    result := arr[0..1] + mid + arr[|arr| - 1..];
}
```

Program → GenAI

When using dafny resolve, the below error is emitted and resolve fails:

```

problem_data/p6_2_d_replace_with_larger_neighbor/program.dfy(8,45): Error: comma expected
|
8 | var mid := seq(i | 1 <= i && i < |arr| - 1 :: if arr[i - 1] >= arr[i + 1] then arr[i
  ↵ - 1] else arr[i + 1]);
  |
  ^

1 parse errors detected in program.dfy

```

Correct the error by altering only the method body. Produce and show only the Dafny body,
 ↵ 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-Co56TVIWvMrb2i7WyKFxxpVPLBTgr

Model: gpt-5.1-chat-latest

Created at: 1766052053

Finish reason: stop

Usage: 981 tokens in, and 161 tokens out

```
{
    var mid := seq(i | 1 <= i && i < |arr| - 1,
                    if arr[i - 1] >= arr[i + 1] then arr[i - 1] else arr[i + 1]);
    result := arr[0..1] + mid + arr[|arr| - 1..];
}
```

Program → GenAI

When using dafny resolve, the below error is emitted and resolve fails:

```
problem_data/p6_2_d_replace_with_larger_neighbor/program.dfy(8,17): Error: unresolved
  ↪ identifier: i
|
8 | var mid := seq(i | 1 <= i && i < |arr| - 1,
| ^

problem_data/p6_2_d_replace_with_larger_neighbor/program.dfy(8,26): Error: unresolved
  ↪ identifier: i
|
8 | var mid := seq(i | 1 <= i && i < |arr| - 1,
| ^

problem_data/p6_2_d_replace_with_larger_neighbor/program.dfy(8,31): Error: unresolved
  ↪ identifier: i
|
8 | var mid := seq(i | 1 <= i && i < |arr| - 1,
| ^

problem_data/p6_2_d_replace_with_larger_neighbor/program.dfy(9,24): Error: unresolved
  ↪ identifier: i
|
9 | if arr[i - 1] >= arr[i + 1] then arr[i - 1] else arr[i + 1]);
| ^

problem_data/p6_2_d_replace_with_larger_neighbor/program.dfy(9,38): Error: unresolved
  ↪ identifier: i
|
9 | if arr[i - 1] >= arr[i + 1] then arr[i - 1] else arr[i + 1]);
| ^

problem_data/p6_2_d_replace_with_larger_neighbor/program.dfy(9,54): Error: unresolved
  ↪ identifier: i
|
9 | if arr[i - 1] >= arr[i + 1] then arr[i - 1] else arr[i + 1]);
| ^

problem_data/p6_2_d_replace_with_larger_neighbor/program.dfy(9,70): Error: unresolved
  ↪ identifier: i
|
9 | if arr[i - 1] >= arr[i + 1] then arr[i - 1] else arr[i + 1]);
```

```

| ^

problem_data/p6_2_d_replace_with_larger_neighbor/program.dfy(8,28): Error: sequence
  ↪ construction must use an integer-based expression for the sequence size (got bool)
|
8 | var mid := seq(i | 1 <= i && i < |arr| - 1,
| ^^

problem_data/p6_2_d_replace_with_larger_neighbor/program.dfy(9,17): Error: sequence-
  ↪ construction initializer expression expected to have type 'nat ~> ?' (instead got
  ↪ 'int')
|
9 | if arr[i - 1] >= arr[i + 1] then arr[i - 1] else arr[i + 1]);
| ^^

problem_data/p6_2_d_replace_with_larger_neighbor/program.dfy(8,19): Error: first argument
  ↪ to | must be of a bitvector type (instead got int)
|
8 | var mid := seq(i | 1 <= i && i < |arr| - 1,
| ^

problem_data/p6_2_d_replace_with_larger_neighbor/program.dfy(8,19): Error: second
  ↪ argument to | must be of a bitvector type (instead got int)
|
8 | var mid := seq(i | 1 <= i && i < |arr| - 1,
| ^

11 resolution/type errors detected in program.dfy

Correct the error by altering only the method body. Produce and show only the Dafny body,
  ↪ 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-Co56X8hqR7w4vJgTPWccafVfWK3Dg

Model: gpt-5.1-chat-latest

Created at: 1766052057

Finish reason: stop

Usage: 2498 tokens in, and 214 tokens out

```
{
  var mid := seq(j | 0 <= j < |arr| - 2 :: if arr[j] >= arr[j + 2] then arr[j] else arr[j
    ↪ + 2]);
  result := [arr[0]] + mid + [arr[|arr| - 1]];
}
```

Program → GenAI

When using dafny resolve, the below error is emitted and resolve fails:

```

problem_data/p6_2_d_replace_with_larger_neighbor/program.dfy(8,40): Error: comma expected
|
8 | var mid := seq(j | 0 <= j < |arr| - 2 :: if arr[j] >= arr[j + 2] then arr[j] else arr
|   ↪ [j + 2]);
|   ^
|
1 parse errors detected in program.dfy

```

Correct the error by altering only the method body. Produce and show only the Dafny body,
 ↪ 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-Co56aU3GgI8ssAlxL1YHaOHrBcVvi

Model: gpt-5.1-chat-latest

Created at: 1766052060

Finish reason: stop

Usage: 2092 tokens in, and 416 tokens out

```
{
  var mid := seq(i | (1 <= i && i < |arr| - 1) :: if arr[i - 1] >= arr[i + 1] then arr[i
    ↪ - 1] else arr[i + 1]);
  result := arr[0..1] + mid + arr[|arr| - 1..];
}
```

Program → GenAI

When using dafny resolve, the below error is emitted and resolve fails:

```

problem_data/p6_2_d_replace_with_larger_neighbor/program.dfy(8,47): Error: comma expected
|
8 | var mid := seq(i | (1 <= i && i < |arr| - 1) :: if arr[i - 1] >= arr[i + 1] then arr[
|   ↪ i - 1] else arr[i + 1]);
|   ^
|
1 parse errors detected in program.dfy

```

Correct the error by altering only the method body. Produce and show only the Dafny body,
 ↪ 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-Co56g3xLJ3W40V4a6YyoyZCPHRGw7

Model: gpt-5.1-chat-latest

Created at: 1766052066

Finish reason: stop

Usage: 2354 tokens in, and 461 tokens out

```
{  
    var s := [arr[0]];  
    var i := 1;  
    while i < |arr| - 1  
        invariant 1 <= i <= |arr| - 1  
        invariant |s| == i  
        invariant s[0] == arr[0]  
        invariant forall k :: 1 <= k < i ==> s[k] == (if arr[k - 1] >= arr[k + 1] then arr[k -  
            ↪ 1] else arr[k + 1])  
    {  
        s := s + [if arr[i - 1] >= arr[i + 1] then arr[i - 1] else arr[i + 1]];  
        i := i + 1;  
    }  
    s := s + [arr[|arr| - 1]];  
    result := s;  
}
```

Final Program

```
method p6_2_d_replace_with_larger_neighbor(arr: seq<int>) returns (result: seq<int>)  
    requires |arr| >= 3  
    ensures |result| == |arr|  
    ensures result[0] == arr[0]  
    ensures result[|arr| - 1] == arr[|arr| - 1]  
    ensures forall i :: 1 <= i < |arr| - 1 ==> result[i] == if arr[i - 1] >= arr[i +  
        ↪ 1] then arr[i - 1] else arr[i + 1]  
{  
    var s := [arr[0]];  
    var i := 1;  
    while i < |arr| - 1  
        invariant 1 <= i <= |arr| - 1  
        invariant |s| == i  
        invariant s[0] == arr[0]  
        invariant forall k :: 1 <= k < i ==> s[k] == (if arr[k - 1] >= arr[k + 1] then arr[k -  
            ↪ 1] else arr[k + 1])  
    {  
        s := s + [if arr[i - 1] >= arr[i + 1] then arr[i - 1] else arr[i + 1]];  
        i := i + 1;  
    }  
    s := s + [arr[|arr| - 1]];  
    result := s;  
}
```

Total Token Usage

Input tokens: 9191

Output tokens: 1696

Reasoning tokens: 1024
Sum of ‘total tokens’: 10887

Experiment Timings

Overall Experiment started at 1766052048133, ended at 1766052074567, lasting 26434ms (26.43 seconds)
Iteration #4 started at 1766052059316, ended at 1766052062445, lasting 3129ms (3.13 seconds)
Iteration #5 started at 1766052062445, ended at 1766052067902, lasting 5457ms (5.46 seconds)
Iteration #6 started at 1766052067902, ended at 1766052074567, lasting 6665ms (6.67 seconds)
Iteration #1 started at 1766052048133, ended at 1766052052752, lasting 4619ms (4.62 seconds)
Iteration #2 started at 1766052052752, ended at 1766052055619, lasting 2867ms (2.87 seconds)
Iteration #3 started at 1766052055619, ended at 1766052059316, lasting 3697ms (3.70 seconds)