

Experiment ‘p6_2_f_move_even_to_front’ Results

December 18, 2025

Experiment outcome: FAILURE_VERIFY

Bad responses: 0

Responses containing assume : 0

Resolution attempts: 8

Hard fails (resolution): 6

Soft fails (resolution): 0

Verification attempts: 2

Problem Specification

Problem name: p6_2_f_move_even_to_front

Natural language statement: Write a method to move all even elements in an array to the front, otherwise preserving the order of the elements.

Method signature: p6_2_f_move_even_to_front(arr: seq<int>) returns (result: seq<int>)

Ensures

- $|result| == |arr|$
- $\forall i, j :: 0 \leq i < j < |result| \ \&\& result[i] \% 2 == 0 \ \&\& result[j] \% 2 != 0$
- $\forall i :: 0 \leq i < |arr| \implies \exists j :: 0 \leq j < |result| \ \&\& arr[i] == result[j]$

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 move all even elements in an array to the front, otherwise preserving
→ the order of the elements.

The signature should be:

```
method p6_2_f_move_even_to_front(arr: seq<int>) returns (result: seq<int>)
```

The method should respect the following contract:

```
ensures |result| == |arr|, ensures forall i, j :: 0 <= i < j < |result| && result[i] % 2  
→ == 0 && result[j] % 2 != 0, ensures forall i :: 0 <= i < |arr| ==> exists j :: 0  
→ <= j < |result| && arr[i] == result[j]
```

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-Co56sbl7J3kmrjeh8sBsG8ey48kU7

Model: gpt-5.1-chat-latest

Created at: 1766052078

Finish reason: stop

Usage: 418 tokens in, and 368 tokens out

```
{  
    var evens: seq<int> := [];  
    var odds: seq<int> := [];  
    var i := 0;  
    while i < |arr|  
        invariant 0 <= i <= |arr|  
        invariant forall k :: 0 <= k < i && arr[k] % 2 == 0 ==> exists j :: 0 <= j < |evens|  
            → && evens[j] == arr[k]  
        invariant forall k :: 0 <= k < i && arr[k] % 2 != 0 ==> exists j :: 0 <= j < |odds| &&  
            → odds[j] == arr[k]  
        invariant |evens| + |odds| == i  
    {  
        if arr[i] % 2 == 0 {  
            evens := evens + [arr[i]];  
        } else {  
            odds := odds + [arr[i]];  
        }  
        i := i + 1;  
    }
```

```

    }
    result := evens + odds;
}

```

Program → GenAI

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

```

problem_data/p6_2_f_move_even_to_front/program.dfy(5,0): Error: a postcondition could not
  ↪ be proved on this return path
|
5 | {
| ^
|   ensures forall i, j :: 0 <= i < j < |result| && result[i] % 2 == 0 && result[j] % 2
|   ↪ != 0
| ^~~~~~
|   ensures forall i :: 0 <= i < |arr| ==> exists j :: 0 <= j < |result| && arr[i] ==
|   ↪ result[j]
| ^~~~~~

problem_data/p6_2_f_move_even_to_front/program.dfy(5,0): Error: a postcondition could not
  ↪ be proved on this return path
|
5 | {
| ^
|   ensures forall i :: 0 <= i < |arr| ==> exists j :: 0 <= j < |result| && arr[i] ==
|   ↪ result[j]
| ^~~~~~

problem_data/p6_2_f_move_even_to_front/program.dfy(4,9): Related location: this is the
  ↪ postcondition that could not be proved
|
4 |   ensures forall i :: 0 <= i < |arr| ==> exists j :: 0 <= j < |result| && arr[i] ==
|   ↪ result[j]
| ^~~~~~

problem_data/p6_2_f_move_even_to_front/program.dfy(11,14): Error: this invariant could
  ↪ not be proved to be maintained by the loop
Related message: loop invariant violation
|
11 | invariant forall k :: 0 <= k < i && arr[k] % 2 == 0 ==> exists j :: 0 <= j < |evens|
  ↪ && evens[j] == arr[k]
| ^~~~~~

problem_data/p6_2_f_move_even_to_front/program.dfy(12,14): Error: this invariant could
  ↪ not be proved to be maintained by the loop
Related message: loop invariant violation
|
12 | invariant forall k :: 0 <= k < i && arr[k] % 2 != 0 ==> exists j :: 0 <= j < |odds|
  ↪ && odds[j] == arr[k]
| ^~~~~~

```

```
Dafny program verifier finished with 1 verified, 4 errors
```

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-Co56xFor6ewDNEXMBviCC6NqDlZzM

Model: gpt-5.1-chat-latest

Created at: 1766052083

Finish reason: stop

Usage: 1490 tokens in, and 157 tokens out

```
{  
    var evens := seq i | 0 <= i < |arr| && arr[i] % 2 == 0 :: arr[i];  
    var odds := seq i | 0 <= i < |arr| && arr[i] % 2 != 0 :: arr[i];  
    result := evens + odds;  
}
```

Program → GenAI

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

```
problem_data/p6_2_f_move_even_to_front/program.dfy(6,19): Error: openparen expected  
|  
6 | var evens := seq i | 0 <= i < |arr| && arr[i] % 2 == 0 :: arr[i];  
| ^  
  
problem_data/p6_2_f_move_even_to_front/program.dfy(6,57): Error: comma expected  
|  
6 | var evens := seq i | 0 <= i < |arr| && arr[i] % 2 == 0 :: arr[i];  
| ^  
  
problem_data/p6_2_f_move_even_to_front/program.dfy(7,19): Error: openparen expected  
|  
7 | var odds := seq i | 0 <= i < |arr| && arr[i] % 2 != 0 :: arr[i];  
| ^  
  
problem_data/p6_2_f_move_even_to_front/program.dfy(7,57): Error: comma expected  
|  
7 | var odds := seq i | 0 <= i < |arr| && arr[i] % 2 != 0 :: arr[i];  
| ^  
  
4 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: chatmpl-Co571e499UUferCG3vFuOzdwHXLnf

Model: gpt-5.1-chat-latest

Created at: 1766052087

Finish reason: stop

Usage: 1725 tokens in, and 223 tokens out

```
{  
    var evens := seq(i | 0 <= i < |arr| && arr[i] % 2 == 0) :: arr[i];  
    var odds := seq(i | 0 <= i < |arr| && arr[i] % 2 != 0) :: arr[i];  
    result := evens + odds;  
}
```

Program → GenAI

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

```
problem_data/p6_2_f_move_even_to_front/program.dfy(6,56): Error: comma expected  
|  
6 | var evens := seq(i | 0 <= i < |arr| && arr[i] % 2 == 0) :: arr[i];  
| ^  
  
problem_data/p6_2_f_move_even_to_front/program.dfy(7,56): Error: comma expected  
|  
7 | var odds := seq(i | 0 <= i < |arr| && arr[i] % 2 != 0) :: arr[i];  
| ^  
  
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: chatmpl-Co574orCHXT48aUMTqf0k6uqHxw6X

Model: gpt-5.1-chat-latest

Created at: 1766052090

Finish reason: stop

Usage: 1895 tokens in, and 95 tokens out

```
{  
    var evens := seq i | (0 <= i < |arr| && arr[i] % 2 == 0) :: arr[i];  
    var odds := seq i | (0 <= i < |arr| && arr[i] % 2 != 0) :: arr[i];  
    result := evens + odds;  
}
```

Program → GenAI

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

```
problem_data/p6_2_f_move_even_to_front/program.dfy(6,19): Error: openparen expected
|
6 | var evens := seq i | (0 <= i < |arr| && arr[i] % 2 == 0) :: arr[i];
| ^

problem_data/p6_2_f_move_even_to_front/program.dfy(6,59): Error: comma expected
|
6 | var evens := seq i | (0 <= i < |arr| && arr[i] % 2 == 0) :: arr[i];
| ^

problem_data/p6_2_f_move_even_to_front/program.dfy(7,19): Error: openparen expected
|
7 | var odds := seq i | (0 <= i < |arr| && arr[i] % 2 != 0) :: arr[i];
| ^

problem_data/p6_2_f_move_even_to_front/program.dfy(7,59): Error: comma expected
|
7 | var odds := seq i | (0 <= i < |arr| && arr[i] % 2 != 0) :: arr[i];
| ^

4 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: chatmpl-Co577ue7pLRJrjYmtEUOCN8olU8dY

Model: gpt-5.1-chat-latest

Created at: 1766052093

Finish reason: stop

Usage: 2451 tokens in, and 285 tokens out

```
{
    var evens := seq(i | 0 <= i < |arr| && arr[i] % 2 == 0) arr[i];
    var odds := seq(i | 0 <= i < |arr| && arr[i] % 2 != 0) arr[i];
    result := evens + odds;
}
```

Program → GenAI

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

```
problem_data/p6_2_f_move_even_to_front/program.dfy(6,56): Error: comma expected
|
```

```

6 | var evens := seq(i | 0 <= i < |arr| && arr[i] % 2 == 0) arr[i];
| ^
problem_data/p6_2_f_move_even_to_front/program.dfy(7,56): Error: comma expected
|
7 | var odds := seq(i | 0 <= i < |arr| && arr[i] % 2 != 0) arr[i];
| ^

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: chatmpl-Co57BM4EAZXwh7bhWTQazOpwXwbMC

Model: gpt-5.1-chat-latest

Created at: 1766052097

Finish reason: stop

Usage: 2611 tokens in, and 159 tokens out

```
{
    var evens := seq(i | 0 <= i < |arr| && arr[i] % 2 == 0) :: arr[i];
    var odds := seq(i | 0 <= i < |arr| && arr[i] % 2 != 0) :: arr[i];
    result := evens + odds;
}
```

Program → GenAI

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

```

problem_data/p6_2_f_move_even_to_front/program.dfy(6,56): Error: comma expected
|
6 | var evens := seq(i | 0 <= i < |arr| && arr[i] % 2 == 0) :: arr[i];
| ^

problem_data/p6_2_f_move_even_to_front/program.dfy(7,56): Error: comma expected
|
7 | var odds := seq(i | 0 <= i < |arr| && arr[i] % 2 != 0) :: arr[i];
| ^

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: chatmpl-Co57EqSZQIGhzhiWEHLFQpM9uLqNm

Model: gpt-5.1-chat-latest

Created at: 1766052100

Finish reason: stop

Usage: 2909 tokens in, and 285 tokens out

```
{  
    var evens := seq(i | 0 <= i < |arr| && arr[i] % 2 == 0 :: arr[i]);  
    var odds := seq(i | 0 <= i < |arr| && arr[i] % 2 != 0 :: arr[i]);  
    result := evens + odds;  
}
```

Program → GenAI

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

```
problem_data/p6_2_f_move_even_to_front/program.dfy(6,57): Error: comma expected  
|  
6 | var evens := seq(i | 0 <= i < |arr| && arr[i] % 2 == 0 :: arr[i]);  
| ^  
  
problem_data/p6_2_f_move_even_to_front/program.dfy(7,57): Error: comma expected  
|  
7 | var odds := seq(i | 0 <= i < |arr| && arr[i] % 2 != 0 :: arr[i]);  
| ^  
  
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: chatmpl-Co57IQ4ut0XKA5DOZhjnEPOUdzol1

Model: gpt-5.1-chat-latest

Created at: 1766052104

Finish reason: stop

Usage: 3199 tokens in, and 446 tokens out

```
{  
    result := [];  
  
    var i := 0;  
    while i < |arr|  
        invariant 0 <= i <= |arr|  
        invariant |result| <= |arr|
```

```

invariant forall k :: 0 <= k < i && arr[k] % 2 == 0 ==> exists j :: 0 <= j < |result|
    ↪ && result[j] == arr[k]
invariant forall j :: 0 <= j < |result| ==> result[j] % 2 == 0
{
if arr[i] % 2 == 0 {
    result := result + [arr[i]];
}
i := i + 1;
}

i := 0;
while i < |arr|
invariant 0 <= i <= |arr|
invariant |result| <= |arr|
invariant forall k :: 0 <= k < |arr| && arr[k] % 2 == 0 ==> exists j :: 0 <= j < |result|
    ↪ && result[j] == arr[k]
invariant forall k :: 0 <= k < i && arr[k] % 2 != 0 ==> exists j :: 0 <= j < |result|
    ↪ && result[j] == arr[k]
invariant forall j :: 0 <= j < |result| ==> (result[j] % 2 == 0 || (exists k :: 0 <= k
    ↪ < i && arr[k] == result[j]))
{
if arr[i] % 2 != 0 {
    result := result + [arr[i]];
}
i := i + 1;
}
}

```

Final Program

```

method p6_2_f_move_even_to_front(arr: seq<int>) returns (result: seq<int>)
ensures |result| == |arr|
ensures forall i, j :: 0 <= i < j < |result| && result[i] % 2 == 0 && result[j] %
    ↪ 2 != 0
ensures forall i :: 0 <= i < |arr| ==> exists j :: 0 <= j < |result| && arr[i] ==
    ↪ result[j]
{
result := [];

var i := 0;
while i < |arr|
invariant 0 <= i <= |arr|
invariant |result| <= |arr|
invariant forall k :: 0 <= k < i && arr[k] % 2 == 0 ==> exists j :: 0 <= j < |result|
    ↪ && result[j] == arr[k]
invariant forall j :: 0 <= j < |result| ==> result[j] % 2 == 0
{
if arr[i] % 2 == 0 {
    result := result + [arr[i]];
}
}

```

```

    i := i + 1;
}

i := 0;
while i < |arr|
  invariant 0 <= i <= |arr|
  invariant |result| <= |arr|
  invariant forall k :: 0 <= k < |arr| && arr[k] % 2 == 0 ==> exists j :: 0 <= j < i
    ↪ result[j] == arr[k]
  invariant forall k :: 0 <= k < i && arr[k] % 2 != 0 ==> exists j :: 0 <= j < |result|
    ↪ && result[j] == arr[k]
  invariant forall j :: 0 <= j < |result| ==> (result[j] % 2 == 0 || (exists k :: 0 <= k
    ↪ < i && arr[k] == result[j]))
{
  if arr[i] % 2 != 0 {
    result := result + [arr[i]];
  }
  i := i + 1;
}
}

```

Total Token Usage

Input tokens: 16698

Output tokens: 2018

Reasoning tokens: 832

Sum of ‘total tokens’: 18716

Experiment Timings

Iteration #8 started at 1766052105913, ended at 1766052112833, lasting 6920ms (6.92 seconds)
Overall Experiment started at 1766052080038, ended at 1766052112833, lasting 32795ms (32.80 seconds)
Iteration #4 started at 1766052092466, ended at 1766052095211, lasting 2745ms (2.75 seconds)
Iteration #5 started at 1766052095211, ended at 1766052099378, lasting 4167ms (4.17 seconds)
Iteration #6 started at 1766052099378, ended at 1766052102092, lasting 2714ms (2.71 seconds)
Iteration #7 started at 1766052102092, ended at 1766052105913, lasting 3821ms (3.82 seconds)
Iteration #1 started at 1766052080039, ended at 1766052085805, lasting 5766ms (5.77 seconds)
Iteration #2 started at 1766052085805, ended at 1766052089096, lasting 3291ms (3.29 seconds)
Iteration #3 started at 1766052089096, ended at 1766052092466, lasting 3370ms (3.37 seconds)