Experiment 'p5_7_countWords' Results

December 18, 2025

Experiment outcome: FAILURE_VERIFY
Bad responses: 0
Responses containing `assume :` 0
Resolution attempts: 8
Hard fails (resolution): 2
Soft fails (resolution): 0
Verification attempts: 6

Problem Specification

Problem name: p5_7_countWords
Natural language statement: Write a method that returns the count of all words in a string. Words are separated by spaces. For example, countWords("Mary had a little lamb") should return 5.
Method signature: p5_7_countWords(str: seq<char>) returns (wordCount: int)

Ensures

- `wordCount >= 0`
- `if |str| == 0 then wordCount == 0 else wordCount >= 1`
- `wordCount == count_space(str) + 1`

Functional Code Given

```
function count_space(str: seq<char>) : int
{
  if |str| == 0 then 0
  else if str[0] == ' ' then 1 + count_space(str[1..])
  else count_space(str[1..])
}
```

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 returns the count of all words in a string. Words are separated by
    ↪ spaces. For example, countWords("Mary had a little lamb") should return 5.

The signature should be:

method p5_7_countWords(str: seq<char>) returns (wordCount: int)

The method should respect the following contract:

ensures wordCount >= 0, ensures if |str| == 0 then wordCount == 0 else wordCount >= 1,
    ↪ ensures wordCount == count_space(str) + 1

The contract uses the following dafny code:

function count_space(str: seq<char>) : int
{
  if |str| == 0 then 0
  else if str[0] == ' ' then 1 + count_space(str[1..])
  else count_space(str[1..])
}

These function(s) must not be used in your implementation of the method.

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-Co4x0Tb24wdz37KG6BopJX1sHz6AH
Model: gpt-5.1-chat-latest
Created at: 1766051466
Finish reason: stop
Usage: 520 tokens in, and 343 tokens out

```
{
  if |str| == 0 {
    wordCount := 0;
  } else {
    var i := 0;
    wordCount := 1;
    while i < |str|
      invariant 0 <= i <= |str|
```

```
    invariant wordCount == count_space(str[..i]) + 1
    invariant wordCount >= 1
    decreases |str| - i
  {
    if str[i] == ' ' {
      wordCount := wordCount + 1;
    }
    i := i + 1;
  }
 }
}
```

Program → GenAI

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

problem_data/p5_7_countWords/program.dfy(12,0): Error: a postcondition could not be
    ↪ proved on this return path
   |
12 | {
   | ^

problem_data/p5_7_countWords/program.dfy(11,19): Related location: this is the
    ↪ postcondition that could not be proved
   |
11 |  ensures wordCount == count_space(str) + 1
   | ^^

problem_data/p5_7_countWords/program.dfy(20,26): Error: this invariant could not be
    ↪ proved to be maintained by the loop
 Related message: loop invariant violation
   |
20 | invariant wordCount == count_space(str[..i]) + 1
   | ^^


Dafny program verifier finished with 1 verified, 2 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: chatcmpl-Co4x6gMze5AIP76VJF4paTZxkuvKO
Model: gpt-5.1-chat-latest
Created at: 1766051472
Finish reason: stop
Usage: 894 tokens in, and 209 tokens out

```
{
  var s := 0;
  var i := 0;
  while i < |str|
    invariant 0 <= i <= |str|
    invariant 0 <= s <= i
    decreases |str| - i
  {
    if str[i] == ' ' {
      s := s + 1;
    }
    i := i + 1;
  }
  if |str| == 0 {
    wordCount := 0;
  } else {
    wordCount := s + 1;
  }
}
```

Program → GenAI

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

problem_data/p5_7_countWords/program.dfy(12,0): Error: a postcondition could not be
    ↪ proved on this return path
   |
12 | {
   | ^

problem_data/p5_7_countWords/program.dfy(11,19): Related location: this is the
    ↪ postcondition that could not be proved
   |
11 |  ensures wordCount == count_space(str) + 1
   |  ^^


Dafny program verifier finished with 1 verified, 1 error

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-Co4xBQd6uD9rrCHrB6xs0RjPj6JpI
Model: gpt-5.1-chat-latest
Created at: 1766051477

Finish reason: stop
Usage: 1150 tokens in, and 423 tokens out

```
{
  // Compute count_space(str) without calling count_space
  var s := 0;
  var i := 0;
  while i < |str|
    invariant 0 <= i <= |str|
    invariant 0 <= s <= i
    invariant s == count_space(str[..i])
    decreases |str| - i
  {
    if str[i] == ' ' {
      s := s + 1;
    }
    i := i + 1;
  }

  // To satisfy the (inconsistent) postconditions, we must return count_space(str) + 1
  wordCount := s + 1;
}
```

Program → GenAI

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

problem_data/p5_7_countWords/program.dfy(12,0): Error: a postcondition could not be
    ↪ proved on this return path
   |
12 | {
   | ^

problem_data/p5_7_countWords/program.dfy(10,38): Related location: this is the
    ↪ postcondition that could not be proved
   |
10 |  ensures if |str| == 0 then wordCount == 0 else wordCount >= 1
   |  ^^

problem_data/p5_7_countWords/program.dfy(12,0): Error: a postcondition could not be
    ↪ proved on this return path
   |
12 | {
   | ^

problem_data/p5_7_countWords/program.dfy(11,19): Related location: this is the
    ↪ postcondition that could not be proved
   |
11 |  ensures wordCount == count_space(str) + 1
   |  ^^
```

```
problem_data/p5_7_countWords/program.dfy(19,16): Error: this invariant could not be
    ↪ proved to be maintained by the loop
 Related message: loop invariant violation
   |
19 | invariant s == count_space(str[..i])
   | ^^


Dafny program verifier finished with 1 verified, 3 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: chatcmpl-Co4xISh1ILm3ivQpb6iVovxb0KJND
Model: gpt-5.1-chat-latest
Created at: 1766051484
Finish reason: stop
Usage: 1803 tokens in, and 495 tokens out

```
{
  // First loop: compute s = number of spaces, no use of count_space in invariants
  var s := 0;
  var i := 0;
  while i < |str|
    invariant 0 <= i <= |str|
    invariant 0 <= s <= i
    decreases |str| - i
  {
    if str[i] == ' ' {
      s := s + 1;
    }
    i := i + 1;
  }

  // Second loop (ghost): prove s == count_space(str)
  ghost var gs := 0;
  ghost var j := 0;
  while j < |str|
    invariant 0 <= j <= |str|
    invariant gs == count_space(str[..j])
    decreases |str| - j
  {
    if str[j] == ' ' {
      gs := gs + 1;
    }
    j := j + 1;
  }
```

```
  // Now gs == count_space(str); and s == gs (same computation)
  assert s == gs;

  if |str| == 0 {
    wordCount := 0;
  } else {
    wordCount := s + 1;
  }
}
```

Program → GenAI

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

problem_data/p5_7_countWords/program.dfy(12,0): Error: a postcondition could not be
    ↪ proved on this return path
   |
12 | {
   | ^

problem_data/p5_7_countWords/program.dfy(11,19): Related location: this is the
    ↪ postcondition that could not be proved
   |
11 |  ensures wordCount == count_space(str) + 1
   | ^^

problem_data/p5_7_countWords/program.dfy(32,17): Error: this invariant could not be
    ↪ proved to be maintained by the loop
 Related message: loop invariant violation
   |
32 | invariant gs == count_space(str[..j])
   | ^^

problem_data/p5_7_countWords/program.dfy(42,2): Error: assertion might not hold
   |
42 | assert s == gs;
   | ^^^^^^


Dafny program verifier finished with 1 verified, 3 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: chatcmpl-Co4xPdcw38nukk22PAzzgCf619zHv

Model: gpt-5.1-chat-latest
Created at: 1766051491
Finish reason: stop
Usage: 2311 tokens in, and 405 tokens out

```
{
  var s := 0;
  var i := 0;
  while i < |str|
    invariant 0 <= i <= |str|
    invariant 0 <= s <= i
    invariant s == count_space(str[..i])
    decreases |str| - i
  {
    if str[i] == ' ' {
      // prove the recursive characterization
      ghost assert count_space(str[..i+1]) == count_space(str[..i]) + 1;
      s := s + 1;
    } else {
      ghost assert count_space(str[..i+1]) == count_space(str[..i]);
    }
    i := i + 1;
  }

  if |str| == 0 {
    wordCount := 0;
  } else {
    wordCount := s + 1;
  }
}
```

Program → GenAI

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

problem_data/p5_7_countWords/program.dfy(23,12): Error: var expected
   |
23 | ghost assert count_space(str[..i+1]) == count_space(str[..i]) + 1;
   | ^

problem_data/p5_7_countWords/program.dfy(26,12): Error: var expected
   |
26 | ghost assert count_space(str[..i+1]) == count_space(str[..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: chatcmpl-Co4xVzZC7mvyvXs4Y3dzk1z66vmYO
Model: gpt-5.1-chat-latest
Created at: 1766051497
Finish reason: stop
Usage: 2603 tokens in, and 204 tokens out

```
{
  var s := 0;
  var i := 0;
  while i < |str|
    invariant 0 <= i <= |str|
    invariant 0 <= s <= i
    invariant s == count_space(str[..i])
    decreases |str| - i
  {
    if str[i] == ' ' {
      assert count_space(str[..i+1]) == count_space(str[..i]) + 1;
      s := s + 1;
    } else {
      assert count_space(str[..i+1]) == count_space(str[..i]);
    }
    i := i + 1;
  }

  if |str| == 0 {
    wordCount := 0;
  } else {
    wordCount := s + 1;
  }
}
```

Program → GenAI

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

problem_data/p5_7_countWords/program.dfy(12,0): Error: a postcondition could not be
    ↪ proved on this return path
   |
12 | {
   | ^

problem_data/p5_7_countWords/program.dfy(11,19): Related location: this is the
    ↪ postcondition that could not be proved
   |
11 |  ensures wordCount == count_space(str) + 1
   |  ^^

problem_data/p5_7_countWords/program.dfy(22,37): Error: assertion might not hold
   |
```

```
22 | assert count_space(str[..i+1]) == count_space(str[..i]) + 1;
   | ^^

problem_data/p5_7_countWords/program.dfy(25,37): Error: assertion might not hold
   |
25 | assert count_space(str[..i+1]) == count_space(str[..i]);
   | ^^


Dafny program verifier finished with 1 verified, 3 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: chatcmpl-Co4xYuv5zE2CBb19wIAEBKxUSzCJW
Model: gpt-5.1-chat-latest
Created at: 1766051500
Finish reason: stop
Usage: 3178 tokens in, and 459 tokens out

```
{
  // First loop: count spaces without linking to count_space
  var s := 0;
  var i := 0;
  while i < |str|
    invariant 0 <= i <= |str|
    invariant 0 <= s <= i
    decreases |str| - i
  {
    if str[i] == ' ' {
      s := s + 1;
    }
    i := i + 1;
  }

  // Second loop (ghost): compute count_space(str)
  ghost var gs := 0;
  ghost var k := 0;
  while k < |str|
    invariant 0 <= k <= |str|
    invariant gs == count_space(str[..k])
    decreases |str| - k
  {
    if str[k] == ' ' {
      gs := gs + 1;
    }
    k := k + 1;
```

```
  }

  if |str| == 0 {
    wordCount := 0;
  } else {
    wordCount := gs + 1;
  }
}
```

Program → GenAI

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

problem_data/p5_7_countWords/program.dfy(44,17): Error: ghost variables such as gs are
    ↪ allowed only in specification contexts. gs was inferred to be ghost based on its
    ↪ declaration or initialization.
   |
44 | wordCount := gs + 1;
   | ^^

1 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: chatcmpl-Co4xfEVfYTgZncsLRwUiDqeRDhxRf
Model: gpt-5.1-chat-latest
Created at: 1766051507
Finish reason: stop
Usage: 3441 tokens in, and 431 tokens out

```
{
  // First loop: compute s = number of spaces (concrete state)
  var s := 0;
  var i := 0;
  while i < |str|
    invariant 0 <= i <= |str|
    invariant 0 <= s <= i
    decreases |str| - i
  {
    if str[i] == ' ' {
      s := s + 1;
    }
    i := i + 1;
  }

  // Second loop (ghost): prove s == count_space(str)
```

```
  ghost var gs := 0;
  ghost var j := 0;
  while j < |str|
    invariant 0 <= j <= |str|
    invariant gs == count_space(str[..j])
    decreases |str| - j
  {
    if str[j] == ' ' {
      gs := gs + 1;
    }
    j := j + 1;
  }

  // Establish equivalence for the postcondition
  assert s == gs;

  // Now use s only (-nonghost)
  if |str| == 0 {
    wordCount := 0;
  } else {
    wordCount := s + 1;
  }
}
```

Final Program

```
function count_space(str: seq<char>) : int
{
  if |str| == 0 then 0
  else if str[0] == ' ' then 1 + count_space(str[1..])
  else count_space(str[1..])
}

method p5_7_countWords(str: seq<char>) returns (wordCount: int)
       ensures wordCount >= 0
       ensures if |str| == 0 then wordCount == 0 else wordCount >= 1
       ensures wordCount == count_space(str) + 1
{
  // First loop: compute s = number of spaces (concrete state)
  var s := 0;
  var i := 0;
  while i < |str|
    invariant 0 <= i <= |str|
    invariant 0 <= s <= i
    decreases |str| - i
  {
    if str[i] == ' ' {
      s := s + 1;
    }
    i := i + 1;
```

```
  }

  // Second loop (ghost): prove s == count_space(str)
  ghost var gs := 0;
  ghost var j := 0;
  while j < |str|
    invariant 0 <= j <= |str|
    invariant gs == count_space(str[..j])
    decreases |str| - j
  {
    if str[j] == ' ' {
      gs := gs + 1;
    }
    j := j + 1;
  }

  // Establish equivalence for the postcondition
  assert s == gs;

  // Now use s only (-nonghost)
  if |str| == 0 {
    wordCount := 0;
  } else {
    wordCount := s + 1;
  }
}
```

Total Token Usage

Input tokens: 15900
Output tokens: 2969
Reasoning tokens: 1216
Sum of 'total tokens': 18869

Experiment Timings

Iteration #8 started at 1766051508850, ended at 1766051515385, lasting 6535ms (6.54 seconds)
Overall Experiment started at 1766051468463, ended at 1766051515385, lasting 46922ms (46.92 seconds)
Iteration #4 started at 1766051486186, ended at 1766051493567, lasting 7381ms (7.38 seconds)
Iteration #5 started at 1766051493567, ended at 1766051498949, lasting 5382ms (5.38 seconds)
Iteration #6 started at 1766051498949, ended at 1766051502700, lasting 3751ms (3.75 seconds)
Iteration #7 started at 1766051502700, ended at 1766051508850, lasting 6150ms (6.15 seconds)
Iteration #1 started at 1766051468463, ended at 1766051474217, lasting 5754ms (5.75 seconds)
Iteration #2 started at 1766051474217, ended at 1766051479168, lasting 4951ms (4.95 seconds)
Iteration #3 started at 1766051479168, ended at 1766051486186, lasting 7018ms (7.02 seconds)