

Experiment ‘p4_10_count_vowels’ Results

December 18, 2025

Experiment outcome: FAILURE_VERIFY

Bad responses: 0

Responses containing assume : 0

Resolution attempts: 8

Hard fails (resolution): 1

Soft fails (resolution): 0

Verification attempts: 7

Problem Specification

Problem name: p4_10_count_vowels

Natural language statement: Write a method that reads a word and returns the number of vowels in the word.

Method signature: p4_10_count_vowels(s: string) returns (count: int)

Ensures

- $0 \leq \text{count}$
- $\text{forall } c :: c \text{ in } s \Rightarrow c \text{ in } ['a', 'e', 'i', 'o', 'u', 'A', 'E', 'I', 'O', 'U'] \Rightarrow \text{count} \geq 1$
- $(\text{forall } c :: c \text{ in } s \Rightarrow c \text{ !in } ['a', 'e', 'i', 'o', 'u', 'A', 'E', 'I', 'O', 'U']) \Rightarrow \text{count} == 0$

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 reads a word and returns the number of vowels in the word.

The signature should be:

method p4_10_count_vowels(s: string) returns (count: int)

The method should respect the following contract:

ensures 0 <= count, ensures forall c :: c in s ==> c in ['a','e','i','o','u','A','E','I',
↪ ', '0','U'] ==> count >= 1, ensures (forall c :: c in s ==> c !in ['a','e','i','o',
↪ ', 'u','A','E','I','O','U']) ==> count == 0

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-Co4ffJAEsy4rXXSQ8WvRkUs0tieVa

Model: gpt-5.1-chat-latest

Created at: 1766050391

Finish reason: stop

Usage: 404 tokens in, and 336 tokens out

```
{
  count := 0;
  var i := 0;
  while i < |s|
    invariant 0 <= count
    invariant 0 <= i <= |s|
    invariant count == |set j:int | 0 <= j < i && (s[j] == 'a' || s[j] == 'e' || s[j] == 'i' || s[j] == 'o' || s[j] == 'u' || s[j] == 'A' || s[j] == 'E' || s[j] == 'I' || s[j] == 'O' || s[j] == 'U')|
  {
    if s[i] == 'a' || s[i] == 'e' || s[i] == 'i' || s[i] == 'o' || s[i] == 'u' || s[i] == 'A' || s[i] == 'E' || s[i] == 'I' || s[i] == 'O' || s[i] == 'U' {
      count := count + 1;
    }
    i := i + 1;
  }
}
```

Program → GenAI

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

```
problem_data/p4_10_count_vowels/program.dfy(5,0): Error: a postcondition could not be
  ↳ proved on this return path
|
5 | {
| ^

problem_data/p4_10_count_vowels/program.dfy(3,9): Related location: this is the
  ↳ postcondition that could not be proved
|
3 | ensures forall c :: c in s ==> c in ['a','e','i','o','u','A','E','I','O','U'] ==>
  ↳ count >= 1
| ~~~~~

problem_data/p4_10_count_vowels/program.dfy(5,0): Error: a postcondition could not be
  ↳ proved on this return path
|
5 | {
| ^

problem_data/p4_10_count_vowels/program.dfy(4,92): Related location: this is the
  ↳ postcondition that could not be proved
|
4 | ensures (forall c :: c in s ==> c !in ['a','e','i','o','u','A','E','I','O','U']) ==>
  ↳ count == 0
| ^^

problem_data/p4_10_count_vowels/program.dfy(11,20): Error: this invariant could not be
  ↳ proved to be maintained by the loop
Related message: loop invariant violation
|
11 | invariant count == |set j:int | 0 <= j < i && (s[j] == 'a' || s[j] == 'e' || s[j] ==
  ↳ 'i' || s[j] == 'o' || s[j] == 'u' || s[j] == 'A' || s[j] == 'E' || s[j] == 'I' ||
  ↳ s[j] == 'O' || s[j] == 'U')|
| ^^

Dafny program verifier finished with 0 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-Co4fBtlwtQDKu7yOkYwIJGCq5u38

Model: gpt-5.1-chat-latest

Created at: 1766050397

Finish reason: stop

Usage: 1425 tokens in, and 550 tokens out

```
{
  count := 0;
  ghost var gcount := 0;
  var i := 0;
  while i < |s|
    invariant 0 <= i <= |s|
    invariant gcount == |set j:int | 0 <= j < i && (s[j] == 'a' || s[j] == 'e' || s[j] ==
      ↪ 'i' || s[j] == 'o' || s[j] == 'u' || s[j] == 'A' || s[j] == 'E' || s[j] == 'I'
      ↪ || s[j] == 'O' || s[j] == 'U')|
    invariant count == gcount
  {
    if s[i] == 'a' || s[i] == 'e' || s[i] == 'i' || s[i] == 'o' || s[i] == 'u' ||
      s[i] == 'A' || s[i] == 'E' || s[i] == 'I' || s[i] == 'O' || s[i] == 'U' {
      count := count + 1;
      gcount := gcount + 1;
    }
    i := i + 1;
  }
}
```

Program → GenAI

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

```
problem_data/p4_10_count_vowels/program.dfy(5,0): Error: a postcondition could not be
  ↪ proved on this return path
```

```
|
5 | {
| ^
```

```
problem_data/p4_10_count_vowels/program.dfy(3,9): Related location: this is the
  ↪ postcondition that could not be proved
```

```
|
3 | ensures forall c :: c in s ==> c in ['a','e','i','o','u','A','E','I','O','U'] ==>
  ↪ count >= 1
| ~~~~~
```

```
problem_data/p4_10_count_vowels/program.dfy(5,0): Error: a postcondition could not be
  ↪ proved on this return path
```

```
|
5 | {
| ^
```

```
problem_data/p4_10_count_vowels/program.dfy(4,92): Related location: this is the
  ↪ postcondition that could not be proved
```

```
|
4 | ensures (forall c :: c in s ==> c !in ['a','e','i','o','u','A','E','I','O','U']) ==>
  ↪ count == 0
```

```
| ^^
problem_data/p4_10_count_vowels/program.dfy(11,21): Error: this invariant could not be
  ↳ proved to be maintained by the loop
Related message: loop invariant violation
|
11 | invariant gcount == |set j:int | 0 <= j < i && (s[j] == 'a' || s[j] == 'e' || s[j]
  ↳ == 'i' || s[j] == 'o' || s[j] == 'u' || s[j] == 'A' || s[j] == 'E' || s[j] == 'I'
  ↳ || s[j] == 'O' || s[j] == 'U')|
| ^^
```

Dafny program verifier finished with 0 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-Co4fs5zP9OGkBFM5wPP77PdxVPExE

Model: gpt-5.1-chat-latest

Created at: 1766050404

Finish reason: stop

Usage: 2188 tokens in, and 279 tokens out

```
{
  count := 0;
  ghost var seenVowel := false;
  var i := 0;
  while i < |s|
    invariant 0 <= i <= |s|
    invariant 0 <= count
    invariant seenVowel ==> count >= 1
    invariant !seenVowel ==> count == 0
  {
    if s[i] == 'a' || s[i] == 'e' || s[i] == 'i' || s[i] == 'o' || s[i] == 'u' ||
      s[i] == 'A' || s[i] == 'E' || s[i] == 'I' || s[i] == 'O' || s[i] == 'U' {
      count := count + 1;
      seenVowel := true;
    }
    i := i + 1;
  }
}
```

Program → GenAI

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

```

problem_data/p4_10_count_vowels/program.dfy(5,0): Error: a postcondition could not be
    ↪ proved on this return path
|
5 | {
| ^

problem_data/p4_10_count_vowels/program.dfy(3,9): Related location: this is the
    ↪ postcondition that could not be proved
|
3 | ensures forall c :: c in s ==> c in ['a','e','i','o','u','A','E','I','O','U'] ==>
    ↪ count >= 1
| ~~~~~

problem_data/p4_10_count_vowels/program.dfy(5,0): Error: a postcondition could not be
    ↪ proved on this return path
|
5 | {
| ^

problem_data/p4_10_count_vowels/program.dfy(4,92): Related location: this is the
    ↪ postcondition that could not be proved
|
4 | ensures (forall c :: c in s ==> c !in ['a','e','i','o','u','A','E','I','O','U']) ==>
    ↪ count == 0
| ^^

Dafny program verifier finished with 0 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-Co4fwd5j7szdnA5XUssvGaNQAIvwn

Model: gpt-5.1-chat-latest

Created at: 1766050408

Finish reason: stop

Usage: 2581 tokens in, and 374 tokens out

```

{
  count := 0;
  ghost var seen := false;
  var i := 0;
  while i < |s|
    invariant 0 <= i <= |s|
    invariant 0 <= count
    invariant seen <==> (exists j:int :: 0 <= j < i && (s[j] == 'a' || s[j] == 'e' || s[j]
      ↪ == 'i' || s[j] == 'o' || s[j] == 'u' || s[j] == 'A' || s[j] == 'E' || s[j] ==
      ↪ 'I' || s[j] == 'O' || s[j] == 'U'))

```

```

    invariant (!seen ==> count == 0)
    invariant (seen ==> count >= 1)
  {
    if s[i] == 'a' || s[i] == 'e' || s[i] == 'i' || s[i] == 'o' || s[i] == 'u' ||
        s[i] == 'A' || s[i] == 'E' || s[i] == 'I' || s[i] == 'O' || s[i] == 'U' {
      count := count + 1;
      seen := true;
    }
    i := i + 1;
  }
}

```

Program → GenAI

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

```

problem_data/p4_10_count_vowels/program.dfy(5,0): Error: a postcondition could not be
    ↪ proved on this return path

```

```

5 | {
  | ^

```

```

problem_data/p4_10_count_vowels/program.dfy(4,92): Related location: this is the
    ↪ postcondition that could not be proved

```

```

4 | ensures (forall c :: c in s ==> c !in ['a','e','i','o','u','A','E','I','O','U']) ==>
    ↪ count == 0
  | ^^

```

Dafny program verifier finished with 0 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-Co4g2x3iiaLcNQ0tRSLzZ95f9cCPU

Model: gpt-5.1-chat-latest

Created at: 1766050414

Finish reason: stop

Usage: 2972 tokens in, and 834 tokens out

```

{
  count := 0;
  var i := 0;
  while i < |s|
    invariant 0 <= i <= |s|
    invariant 0 <= count

```

```

{
  if s[i] == 'a' || s[i] == 'e' || s[i] == 'i' || s[i] == 'o' || s[i] == 'u' ||
    s[i] == 'A' || s[i] == 'E' || s[i] == 'I' || s[i] == 'O' || s[i] == 'U' {
    count := count + 1;
  }
  i := i + 1;
}

ghost var noVowel := true;
ghost var j := 0;
while j < |s|
  invariant 0 <= j <= |s|
  invariant noVowel ==> (forall c :: c in s[..j] ==> c !in ['a','e','i','o','u','A','E'
    ↪ ', 'I', 'O', 'U'])
  {
    if s[j] == 'a' || s[j] == 'e' || s[j] == 'i' || s[j] == 'o' || s[j] == 'u' ||
      s[j] == 'A' || s[j] == 'E' || s[j] == 'I' || s[j] == 'O' || s[j] == 'U' {
      noVowel := false;
    }
    j := j + 1;
  }

  if noVowel {
    count := 0;
  }
}

```

Program → GenAI

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

```

problem_data/p4_10_count_vowels/program.dfy(33,10): Error: assignment to non-ghost
  ↪ variable is not allowed in this context, because the statement is in a ghost
  ↪ context; e.g., it may be guarded by a specification-only expression
|
33 | count := 0;
| ^^

```

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-Co4gB1pHVB5FgVnNt1z0YnBswfMyG

Model: gpt-5.1-chat-latest

Created at: 1766050423

Finish reason: stop

Usage: 3440 tokens in, and 273 tokens out

```
{
  count := 0;
  var seen := false;
  var i := 0;
  while i < |s|
    invariant 0 <= i <= |s|
    invariant 0 <= count
    invariant (!seen ==> count == 0)
    invariant (seen ==> count >= 1)
  {
    if s[i] == 'a' || s[i] == 'e' || s[i] == 'i' || s[i] == 'o' || s[i] == 'u' ||
      s[i] == 'A' || s[i] == 'E' || s[i] == 'I' || s[i] == 'O' || s[i] == 'U' {
      count := count + 1;
      seen := true;
    }
    i := i + 1;
  }
}
```

Program → GenAI

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

```
problem_data/p4_10_count_vowels/program.dfy(5,0): Error: a postcondition could not be
  ↳ proved on this return path
```

```
|
5 | {
| ^
```

```
problem_data/p4_10_count_vowels/program.dfy(3,9): Related location: this is the
  ↳ postcondition that could not be proved
```

```
|
3 | ensures forall c :: c in s ==> c in ['a','e','i','o','u','A','E','I','O','U'] ==>
  ↳ count >= 1
| ~~~~~
```

```
problem_data/p4_10_count_vowels/program.dfy(5,0): Error: a postcondition could not be
  ↳ proved on this return path
```

```
|
5 | {
| ^
```

```
problem_data/p4_10_count_vowels/program.dfy(4,92): Related location: this is the
  ↳ postcondition that could not be proved
```

```
|
4 | ensures (forall c :: c in s ==> c !in ['a','e','i','o','u','A','E','I','O','U']) ==>
  ↳ count == 0
| ^^
```

Dafny program verifier finished with 0 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-Co4gG0yI1dR5WQOHiltWt5WbGRWq4

Model: gpt-5.1-chat-latest

Created at: 1766050428

Finish reason: stop

Usage: 4153 tokens in, and 424 tokens out

```
{
  count := 0;
  var i := 0;
  while i < |s|
    invariant 0 <= i <= |s|
    invariant 0 <= count
  {
    if s[i] == 'a' || s[i] == 'e' || s[i] == 'i' || s[i] == 'o' || s[i] == 'u' ||
      s[i] == 'A' || s[i] == 'E' || s[i] == 'I' || s[i] == 'O' || s[i] == 'U' {
      count := count + 1;
    }
    i := i + 1;
  }

  ghost var existsVowel := false;
  ghost var j := 0;
  while j < |s|
    invariant 0 <= j <= |s|
    invariant existsVowel ==> true
  {
    if s[j] == 'a' || s[j] == 'e' || s[j] == 'i' || s[j] == 'o' || s[j] == 'u' ||
      s[j] == 'A' || s[j] == 'E' || s[j] == 'I' || s[j] == 'O' || s[j] == 'U' {
      existsVowel := true;
    }
    j := j + 1;
  }

  assert existsVowel ==> count >= 1;
  assert (!existsVowel) ==> count == 0;
}
```

Program → GenAI

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

```

problem_data/p4_10_count_vowels/program.dfy(5,0): Error: a postcondition could not be
    ↪ proved on this return path
|
5 | {
| ^

problem_data/p4_10_count_vowels/program.dfy(3,9): Related location: this is the
    ↪ postcondition that could not be proved
|
3 | ensures forall c :: c in s ==> c in ['a','e','i','o','u','A','E','I','O','U'] ==>
    ↪ count >= 1
| ~~~~~

problem_data/p4_10_count_vowels/program.dfy(5,0): Error: a postcondition could not be
    ↪ proved on this return path
|
5 | {
| ^

problem_data/p4_10_count_vowels/program.dfy(4,92): Related location: this is the
    ↪ postcondition that could not be proved
|
4 | ensures (forall c :: c in s ==> c !in ['a','e','i','o','u','A','E','I','O','U']) ==>
    ↪ count == 0
| ^^

problem_data/p4_10_count_vowels/program.dfy(32,31): Error: assertion might not hold
|
32 | assert existsVowel ==> count >= 1;
| ^^

problem_data/p4_10_count_vowels/program.dfy(33,34): Error: assertion might not hold
|
33 | assert (!existsVowel) ==> count == 0;
| ^^

Dafny program verifier finished with 0 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: chatcmpl-Co4gNRucaCd5QS5TXGSxbzqs4ctHl

Model: gpt-5.1-chat-latest

Created at: 1766050435

Finish reason: stop

Usage: 5012 tokens in, and 312 tokens out

```
{
  count := 0;
  ghost var vowelIdxs : set<int> := {};
  var i := 0;
  while i < |s|
    invariant 0 <= i <= |s|
    invariant 0 <= count
    invariant vowelIdxs == set j:int | 0 <= j < i && (s[j] == 'a' || s[j] == 'e' || s[j]
      ↪ == 'i' || s[j] == 'o' || s[j] == 'u' || s[j] == 'A' || s[j] == 'E' || s[j] == '
      ↪ I' || s[j] == 'O' || s[j] == 'U')
    invariant count == |vowelIdxs|
    {
      if s[i] == 'a' || s[i] == 'e' || s[i] == 'i' || s[i] == 'o' || s[i] == 'u' ||
        s[i] == 'A' || s[i] == 'E' || s[i] == 'I' || s[i] == 'O' || s[i] == 'U' {
        vowelIdxs := vowelIdxs + {i};
        count := count + 1;
      }
      i := i + 1;
    }
  }
}
```

Final Program

```
method p4_10_count_vowels(s: string) returns (count: int)
  ensures 0 <= count
  ensures forall c :: c in s ==> c in ['a','e','i','o','u','A','E','I','O','U'] ==>
    ↪ count >= 1
  ensures (forall c :: c in s ==> c !in ['a','e','i','o','u','A','E','I','O','U'])
    ↪ ==> count == 0
{
  count := 0;
  ghost var vowelIdxs : set<int> := {};
  var i := 0;
  while i < |s|
    invariant 0 <= i <= |s|
    invariant 0 <= count
    invariant vowelIdxs == set j:int | 0 <= j < i && (s[j] == 'a' || s[j] == 'e' || s[j]
      ↪ == 'i' || s[j] == 'o' || s[j] == 'u' || s[j] == 'A' || s[j] == 'E' || s[j] == '
      ↪ I' || s[j] == 'O' || s[j] == 'U')
    invariant count == |vowelIdxs|
    {
      if s[i] == 'a' || s[i] == 'e' || s[i] == 'i' || s[i] == 'o' || s[i] == 'u' ||
        s[i] == 'A' || s[i] == 'E' || s[i] == 'I' || s[i] == 'O' || s[i] == 'U' {
        vowelIdxs := vowelIdxs + {i};
        count := count + 1;
      }
      i := i + 1;
    }
  }
}
```

Total Token Usage

Input tokens: 22175

Output tokens: 3382

Reasoning tokens: 1024

Sum of ‘total tokens’: 25557

Experiment Timings

Iteration #8 started at 1766050436976, ended at 1766050442271, lasting 5295ms (5.30 seconds)

Overall Experiment started at 1766050393566, ended at 1766050442271, lasting 48705ms (48.71 seconds)

Iteration #4 started at 1766050410410, ended at 1766050416052, lasting 5642ms (5.64 seconds)

Iteration #5 started at 1766050416052, ended at 1766050425587, lasting 9535ms (9.54 seconds)

Iteration #6 started at 1766050425587, ended at 1766050430649, lasting 5062ms (5.06 seconds)

Iteration #7 started at 1766050430649, ended at 1766050436976, lasting 6327ms (6.33 seconds)

Iteration #1 started at 1766050393567, ended at 1766050398975, lasting 5408ms (5.41 seconds)

Iteration #2 started at 1766050398975, ended at 1766050406056, lasting 7081ms (7.08 seconds)

Iteration #3 started at 1766050406056, ended at 1766050410410, lasting 4354ms (4.35 seconds)