

Experiment ‘P_3_2_classify_real’ Results

December 18, 2025

Experiment outcome: SUCCESS

Bad responses: 0

Responses containing assume : 0

Resolution attempts: 2

Hard fails (resolution): 1

Soft fails (resolution): 0

Verification attempts: 1

Problem Specification

Problem name: P_3_2_classify_real

Natural language statement: Write a method that takes a floating point number and returns whether it is negative, zero, or positive. Add "small" if the absolute value of the number is less than 1, or "large" if it exceeds 1,000,000.

Method signature: P_3_2_classify_real (x:real) returns (s: string)

Ensures

- $x == 0.0 \implies s == \text{"zero, small"}$
- $1.0 > x > 0.0 \implies s == \text{"positive, small"}$
- $1.0 \leq x < 1000000.0 \implies s == \text{"positive"}$
- $1000000.0 \leq x \implies s == \text{"positive, large"}$
- $0.0 > x > -1.0 \implies s == \text{"negative, small"}$
- $-1.0 \geq x > -1000000.0 \implies s == \text{"negative"}$
- $-1000000.0 \geq x \implies s == \text{"negative, large"}$

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 takes a floating point number and returns whether it is negative,
→ zero, or positive. Add "small" if the absolute value of the number is less than 1,
→ or "large" if it exceeds 1,000,000.

The signature should be:

```
method P_3_2_classify_real (x:real) returns (s: string)
```

The method should respect the following contract:

```
ensures x == 0.0 ==> s == "zero, small", ensures 1.0 > x > 0.0 ==> s == "positive, small  
→ ", ensures 1.0 <= x < 1000000.0 ==> s == "positive", ensures 1000000.0 <= x ==> s  
→ == "positive, large", ensures 0.0 > x > -1.0 ==> s == "negative, small", ensures  
→ -1.0 >= x > -1000000.0 ==> s == "negative", ensures -1000000.0 >= x ==> s == "  
→ negative, large"
```

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-Co4S4PPipSMR3cI8b4TTuZMdM0tj0

Model: gpt-5.1-chat-latest

Created at: 1766049548

Finish reason: stop

Usage: 574 tokens in, and 155 tokens out

```
{  
  if x == 0.0 then  
    s := "zero, small"  
  else if x > 0.0 then  
    if x < 1.0 then  
      s := "positive, small"  
    else if x < 1000000.0 then  
      s := "positive"  
    else  
      s := "positive, large"  
  else  
    if x > -1.0 then  
      s := "negative, small"  
    else if x > -1000000.0 then  
      s := "negative"
```

```

    else
      s := "negative, large"
    }

```

Program → GenAI

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

```

problem_data/P_3_2_classify_real/program.dfy(10,14): Error: lbrace expected
|
10 | if x == 0.0 then
|   ^

```

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-Co4S9PN2KCVm6OS5cTwymg8zZ0V2i

Model: gpt-5.1-chat-latest

Created at: 1766049553

Finish reason: stop

Usage: 691 tokens in, and 180 tokens out

```

{
  if x == 0.0 {
    s := "zero, small";
  } else if x > 0.0 {
    if x < 1.0 {
      s := "positive, small";
    } else if x < 1000000.0 {
      s := "positive";
    } else {
      s := "positive, large";
    }
  } else {
    if x > -1.0 {
      s := "negative, small";
    } else if x > -1000000.0 {
      s := "negative";
    } else {
      s := "negative, large";
    }
  }
}

```

Final Program

```
method P_3_2_classify_real (x:real) returns (s: string)
  ensures x == 0.0 ==> s == "zero, small"
  ensures 1.0 > x > 0.0 ==> s == "positive, small"
  ensures 1.0 <= x < 1000000.0 ==> s == "positive"
  ensures 1000000.0 <= x ==> s == "positive, large"
  ensures 0.0 > x > -1.0 ==> s == "negative, small"
  ensures -1.0 >= x > -1000000.0 ==> s == "negative"
  ensures -1000000.0 >= x ==> s == "negative, large"
{
  if x == 0.0 {
    s := "zero, small";
  } else if x > 0.0 {
    if x < 1.0 {
      s := "positive, small";
    } else if x < 1000000.0 {
      s := "positive";
    } else {
      s := "positive, large";
    }
  } else {
    if x > -1.0 {
      s := "negative, small";
    } else if x > -1000000.0 {
      s := "negative";
    } else {
      s := "negative, large";
    }
  }
}
```

Total Token Usage

Input tokens: 1265

Output tokens: 335

Reasoning tokens: 0

Sum of 'total tokens': 1600

Experiment Timings

Overall Experiment started at 1766049549817, ended at 1766049559160, lasting 9343ms (9.34 seconds)

Iteration #1 started at 1766049549818, ended at 1766049555292, lasting 5474ms (5.47 seconds)

Iteration #2 started at 1766049555292, ended at 1766049559160, lasting 3868ms (3.87 seconds)