

# Homework 1: Reasoning About Code

*Due: Friday, Feb. 2, 2024, 11:59:59 pm*

## Submission Instructions

- This homework is different from the previous homework. It does not involve any Java coding. Rather, you will answer some questions involving reasoning about code.
- Follow the directions in the [version control handout](#) for cloning your hw01 git repo. The URI that you will need to use to clone your personal repo for this homework would have the form of `https://submitty.cs.rpi.edu/git/s24/csci2600/hw01/RCSID` where RCSID is your RCS ID. Submit your answers in a single .PDF file named `hw1.answers.pdf` in the `answers/` directory of your repository. **You MUST type up your answers. Handwritten solutions will not be accepted or graded, even if they are scanned into a PDF file.** We recommend using [LaTeX](#). If you have never used LaTeX, take a look at this [tutorial](#).
- Be sure to commit and push the file to Submittity. Follow the directions in the [version control handout](#) for adding and committing files.
- **You must click the [Grade My Repository](#) button for you answers to be graded. If you do not, they will not be graded and you will receive a zero for the homework.**

### Directions:

- Unless otherwise indicated, assume that all referenced numerical variables are defined as integers, and that integer overflow or underflow will not occur.
- In your answers, please use standard logical symbols  $\wedge$  and  $\vee$  for “and” and “or” respectively. Use  $=$  for equivalence (comparing equality) in logical statements.
- Note that braces (`{` and `}`) are used in this homework to enclose conditions, as well as to define blocks of code. Make sure you distinguish between these two different uses of braces based on the context.
- If no precondition is required for a code sequence, simply write `{_true_}` to denote the trivial precondition.
- You must **show work** for all problems in this homework in order to get credit. If you provide just the final answer but fail to show work, you will get a zero, even if the answer is correct.
- All code is in Java and assumes the use of standard Java library.

## Problems

### Problem 1 (12 pts., 2 pts. each): Condition Strength

Order conditions from the **weakest to the strongest** by showing the longest implication chain of the form  $A \rightarrow B \rightarrow C \rightarrow D \dots$  (where  $A, B, C, D, \dots$  are given conditions,  $\rightarrow$  is implication). If total ordering by strength cannot be established, show a partial ordering. Prove by showing all necessary implications. If any implication is false, provide a counterexample. You must show all work in order to get full credit. Unless otherwise stated, assume all referenced variables are defined as integers.

- (1)  $\{ x = 2k + 1 \wedge x = x + 1 \wedge y = 10 \}$   
 $\{ x \text{ is divisible by } 6 \wedge y = x + 2 \}$   
 $\{ x = 18 \wedge y = 20 \}$   
 $\{ x \% 2 = 0 \wedge y \text{ is even} \}$
- (2)  $\{ 5 \leq k < 5 \}$   
 $\{ k \geq -10 \}$   
 $\{ -10 < k \leq 1 \}$   
 $\{ 10 \leq k \leq -10 \}$
- (3)  $\{ x \geq 0 \wedge y > x \}$   
 $\{ x = 3 \wedge y > 10 \}$   
 $\{ x = y \% 10 \}$   
 $\{ xy = 0 \}$   
 $\{ x > 1 \vee y > 1 \}$
- (4) Recall the names for infinite sets of numbers used in math (e.g., see “Peter Jephson Cameron (1998). *Introduction to Algebra*. Oxford University Press. p. 4. ISBN 978-0-19-850195-4.”)  
 $\{ z \in \mathbb{Q} \}$   
 $\{ z = \sqrt{-1} \}$   
 $\{ z \in \mathbb{N} \}$   
 $\{ z = m / n \}$   
 $\{ z \in \mathbb{Z} \}$   
 $\{ z = y \% 2 \}$   
 $\{ z \in \mathbb{R} \}$   
 $\{ z = y / 2 \}$
- (5)  $\{ -7 < x < 0 \}$   
 $\{ y = \log(x) \}$   
 $\{ -5 < x \leq 10 \}$   
 $\{ x = 1 \wedge y \geq 1 \}$   
 $\{ -1 \leq x \leq 1 \}$
- (6) Assume "result is a double" and  $|\cdot|$  is the absolute value operator.  
 $\{ |\text{result} - \sin(x)| \leq 1 \}$   
 $\{ |\text{result} - \sin(x)| \leq 0.01 \}$   
 $\{ |\text{result} - \sin(x)| \leq 10^{-10} \}$   
 $\{ |\text{result} - \sin(x)| \leq -0.01 \}$   
 $\{ |\text{result} - \sin(x)| \geq x^2 \}$

**Problem 2 (4 pts., 1 pt. each): Hoare Triples**

State whether each Hoare triple is valid. If it is invalid, explain why and show how you would modify the the postcondition to make it valid. Unless otherwise stated, assume all referenced variables are defined as integers.

(1)  $\{ x = 5 \}$

$x = x * 2;$

$\{ x = 10 \vee x \neq 0 \}$

(2)  $\{ \sqrt{x-1} > k \}$

$x = x + 1;$

$\{ k \geq 0 \}$

(3)  $\{ i + j \neq 0 \wedge i \cdot j = 0 \}$

$i = j - 1;$

$j = i + 1;$

$\{ (i = 0 \vee i \neq -j) \wedge k \in \mathbb{Q} \}$

(4)  $\{ n < 0 \wedge n = \sqrt{m} \}$

**if**  $(n < m)$

$x = n;$

**else**

$y = m;$

$\{ x \neq y \}$

### Problem 3 (4 pts., 1 pt. each): General Hoare Triples

A, B, C, D, E, and F are logical conditions (logical formulas). The following are true:

- $A \rightarrow B$  (A implies B, i.e., A is stronger than B)
- $B \rightarrow C$
- $C \rightarrow B$
- $D \rightarrow E$
- $E \rightarrow F$
- $\{ B \} \text{ code } \{ E \}$

Indicate whether the following Hoare triples are valid or possibly invalid. If possibly invalid, prove by giving an example.

(1)  $\{ C \} \text{ code } \{ D \}$

(2)  $\{ B \} \text{ code } \{ C \}$

(3)  $\{ A \} \text{ code } \{ D \}$

(4)  $\{ A \} \text{ code } \{ F \}$

### Problem 4 (8 pts., 1 pt. for each condition): Forward reasoning

For each code snippet with the given precondition, find the **strongest postcondition** by inserting the appropriate condition in each blank. The first intermediate condition in part (1) is supplied as an example. Please simplify your answers as much as possible. Assume all referenced variables are defined as integers. Copy all code to your answer file and fill in the blanks. Carry all variables forward. Show all work.

(1)  $\{ z \neq 0 \}$   
   $y = 0;$   
     $\{ y = 10 \wedge z \neq 0 \}$   
   $x = y + 2;$   
     $\{ \text{-----} \}$   
   $z = x + y;$   
     $\{ \text{-----} \}$

(2)  $\{ |x| > 5 \}$   
   $x = x \% 10;$   
     $\{ \text{-----} \}$   
   $x = x * x;$   
     $\{ \text{-----} \}$   
   $x = -x;$   
     $\{ \text{-----} \}$

```

(3) { z < 5 }
    if (z > 0) {
        {-----}
    }
    z = -z;
    {-----}
}
{-----}

```

**Problem 5 (11 pts., 0.5 pts. each condition): Backward reasoning**

Find the **weakest precondition** of each code sequence by inserting the appropriate condition in each blank. The first intermediate condition in part (1) is supplied as an example. Please simplify your answers as much as possible. Assume all referenced variables are defined as integers. Use the `wp()` notation as shown in the first example.

```

(1) {-----}
    x = -1;
    { wp("z = 3 * x + y;", 0 < z < 10) } = { 0 < 3x + y < 10 } =
    { y > -3x ∧ y < 10 - 3x }
    z = 3 * x + y;
    { 0 < z < 10 }

```

```

(2) {-----}
    if (y > 0) {
        {-----}
        x = x / y;
        {-----}
        y = 0;
        {-----}
    } else {
        {-----}
        y = 4 * x;
        {-----}
    }
    { x > 1 ∧ y = 0 }

```

```

(3) {-----}
    if (x ≥ 0) {
        {-----}
        z = Math.min(z, x);
        {-----}
        x = z + y;
        {-----}
    }
    { z ≠ 0 ∧ y ≥ 0 ∨ x ≥ 0 }

```

(4) {-----}

```

if (Math.abs(x) <= 5) {
    {-----}
    z = x - 2;
    {-----}
} else {
    {-----}
    if (x <= -5) {
        {-----}
        z = x + 6;
        {-----}
    } else {
        {-----}
        z = 3 * x;
        {-----}
    }
    {-----}
}
{ -3 ≤ z ≤ 3 }

```

(5) {-----}

```

x = y / 2;
{-----}
z = x + 1;
{ x ≠ 0.5 ∧ z > 0 }

```

**Problem 6 (10 pts., 1 pt. each condition, 1 pt. sufficient/insufficient): Verifying Correctness**

For each block of code, fill in all the conditions, then use them to state whether the precondition is sufficient to guarantee the postcondition. If the precondition is insufficient, explain why.

Hint: Use backward reasoning to find the weakest precondition that guarantees the postcondition and see if the given precondition is strong enough to guarantee the postcondition. In other words, is the given precondition not stronger than the weakest precondition?

Copy all code to your answer file and fill in the blanks. Answers must be expressed in the format `wp("code", condition) = precondition`. Show all work. Assume all referenced variables are defined as integers.

(1) {  $x < 2$  }  
    {-----}  
     $z = x - z$ ;  
    {-----}  
     $w = x - 1$ ;  
    {-----}  
     $z = w - 1$ ;  
    { $z > 1$ }  
    Sufficient or Insufficient: -----

(2) {  $x = y \wedge y > 0 \vee y \neq x \wedge x \geq 0$  }  
    {-----}  
    if ( $x < y$ ) {  
        {-----}  
         $x--$ ;  
        {-----}  
    } else {  
        {-----}  
         $x = y / 2$ ;  
        {-----}  
    }  
    {  $x < y \wedge x \geq 0$  }  
    Sufficient or Insufficient: -----

## Collaboration (0.5 pts)

Please answer the following questions in a file named `collaboration.pdf` in your `answers/` directory.

The standard [academic integrity policy](#) applies to this homework.

State whether or not you collaborated with other students. If you did collaborate with other students, state their names and a brief description of how you collaborated.

## Reflection (0.5 pts)

Please answer the following questions in a file named `reflection.pdf` in your `answers/` directory. Answer briefly, but in enough detail to help you improve your own practice via introspection and to enable me to improve Principles of Software in the future.

- In retrospect, what could you have done better to reduce the time you spent solving this homework?
- What could we, the teaching staff, have done better to improve your learning experience in this homework?
- What do you know now that you did not know before beginning the homework?

## Errata

Check the [Submittity Forum](#) for possible errata.