Invariants

Vladimir Podolskii

Computer Science Department, Higher School of Economics

Outline

Invariants

Coffee with milk

More Coffee

Debugging Problem

Invariants

- Looking at the right property is important in general

Invariants

- Looking at the right property is important in general
- Sometimes it is a number, sometimes it is something else

Invariants

- Looking at the right property is important in general
- Sometimes it is a number, sometimes it is something else
- The properties that do not change are called invariants

Invariants

- Looking at the right property is important in general
- Sometimes it is a number, sometimes it is something else
- The properties that do not change are called invariants
- Double counting is a special case

Outline

Invariants

Coffee with milk

More Coffee

Debugging Problem

Coffee with milk

Problem

There are two cups, one with coffee and another with milk. We take a spoon of coffee and add it to the cup with milk. After that we take one spoon of a drink in the second cup and put it to the first cup. What is larger, the amount of milk in the coffee cup or the amount of coffee in the milk cup?

- It might seem that some serious calculations are needed

Coffee with milk

Problem

There are two cups, one with coffee and another with milk. We take a spoon of coffee and add it to the cup with milk. After that we take one spoon of a drink in the second cup and put it to the first cup. What is larger, the amount of milk in the coffee cup or the amount of coffee in the milk cup?

- It might seem that some serious calculations are needed
- We don't know the sizes of cups and the size of the spoon

Coffee with milk

Problem

There are two cups, one with coffee and another with milk. We take a spoon of coffee and add it to the cup with milk. After that we take one spoon of a drink in the second cup and put it to the first cup. What is larger, the amount of milk in the coffee cup or the amount of coffee in the milk cup?

- It might seem that some serious calculations are needed
- We don't know the sizes of cups and the size of the spoon
- Maybe the result depends on these parameters?

Coffee with milk

Problem

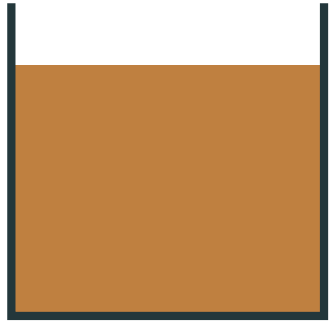
There are two cups, one with coffee and another with milk. We take a spoon of coffee and add it to the cup with milk. After that we take one spoon of a drink in the second cup and put it to the first cup. What is larger, the amount of milk in the coffee cup or the amount of coffee in the milk cup?

- It might seem that some serious calculations are needed
- We don't know the sizes of cups and the size of the spoon
- Maybe the result depends on these parameters?
- It turns out that no calculations are needed!

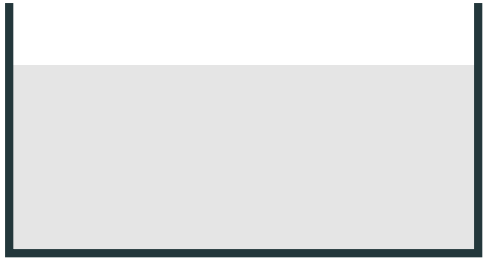
Coffee with milk

Before

Cup 1

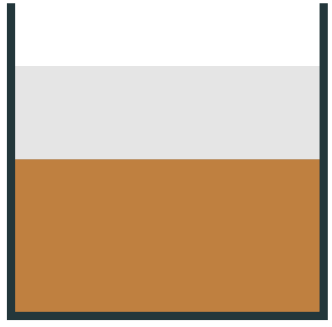


Cup 2



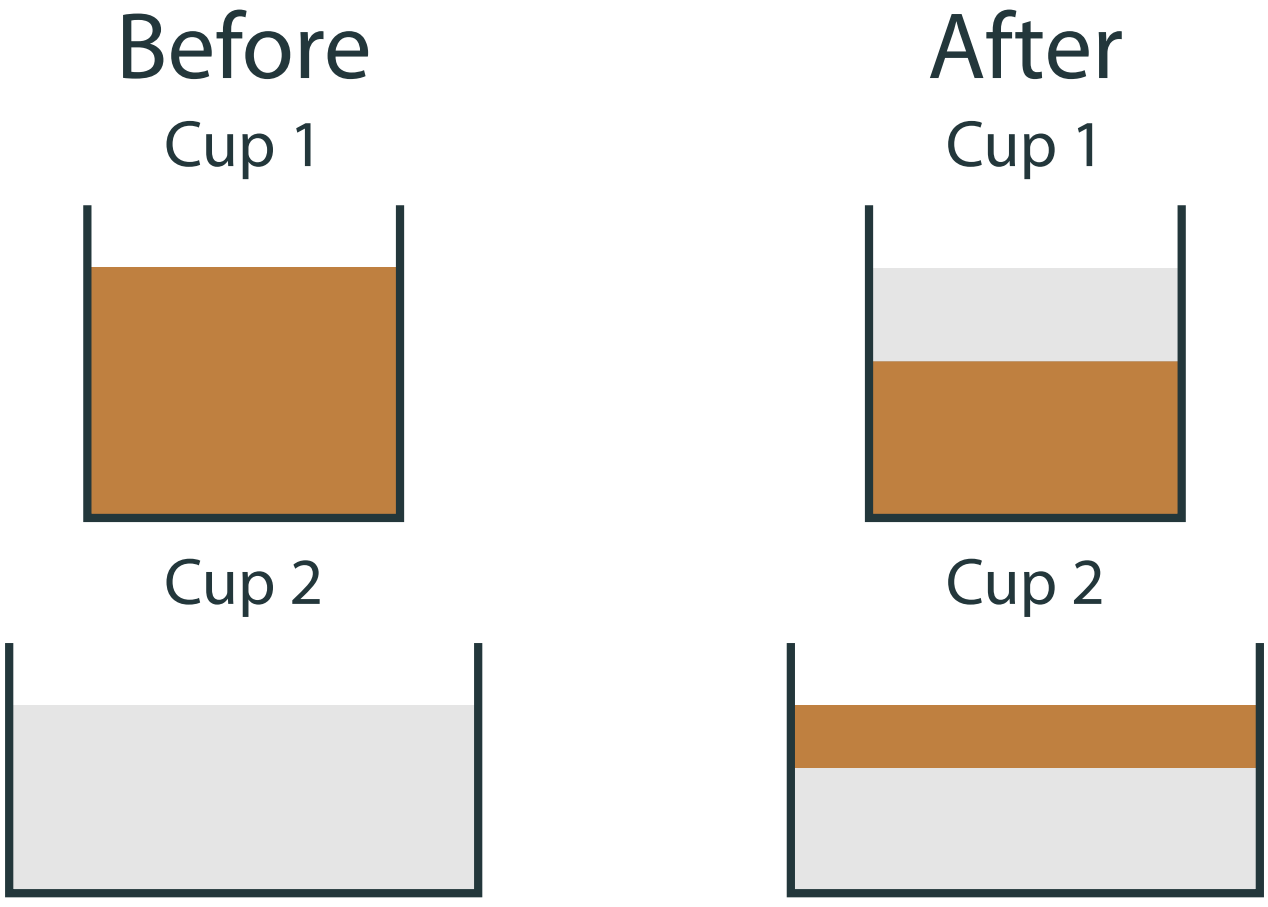
After

Cup 1



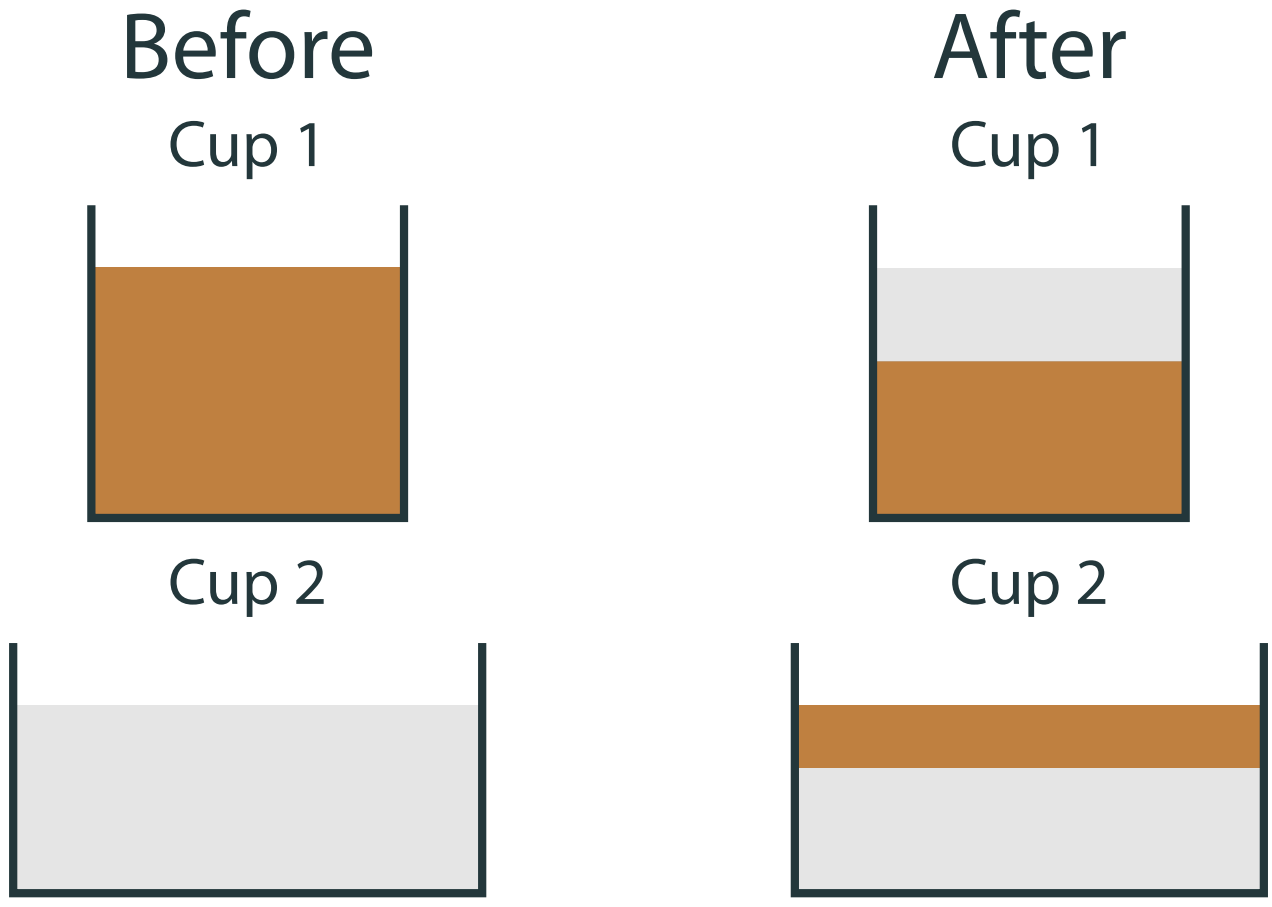
Cup 2

Coffee with milk



The size of a drink in Cup 1 is invariant!

Coffee with milk



The size of a drink in Cup 1 is invariant!

So the amount of coffee missing in Cup 1 is the same as the amount of milk added to Cup 1!

Outline

Invariants

Coffee with milk

More Coffee

Debugging Problem

More Coffee

Problem

There are two equally sized cups, one with coffee, another with milk. Both cups are half full. We like the first cup and we like coffee with lots of milk: $1/3$ coffee, $2/3$ milk. We can pour drinks from one cup to another back and forth. Can we get our favorite coffee in our favorite cup? Any amount would do, the right proportion is what matters.

- The previous problem does not help: it is fine if the second cup is mostly coffee

More Coffee

Problem

There are two equally sized cups, one with coffee, another with milk. Both cups are half full. We like the first cup and we like coffee with lots of milk: $1/3$ coffee, $2/3$ milk. We can pour drinks from one cup to another back and forth. Can we get our favorite coffee in our favorite cup? Any amount would do, the right proportion is what matters.

- The previous problem does not help: it is fine if the second cup is mostly coffee
- Yet, invariants can help again

More Coffee

Problem

There are two equally sized cups, one with coffee, another with milk. Both cups are half full. We like the first cup and we like coffee with lots of milk: $1/3$ coffee, $2/3$ milk. We can pour drinks from one cup to another back and forth. Can we get our favorite coffee in our favorite cup? Any amount would do, the right proportion is what matters.

- The previous problem does not help: it is fine if the second cup is mostly coffee
- Yet, invariants can help again
- Just have to choose the right invariant

More Coffee

Claim

The proportion of coffee in the first cup is always greater than in the second cup. That is, coffee in the first cup is stronger.

- Inequality between proportions of coffee in cups is our invariant

More Coffee

Claim

The proportion of coffee in the first cup is always greater than in the second cup. That is, coffee in the first cup is stronger.

- Inequality between proportions of coffee in cups is our invariant
- It does not allow us to have more milk than coffee in the first cup

More Coffee

Claim

The proportion of coffee in the first cup is always greater than in the second cup. That is, coffee in the first cup is stronger.

- Inequality between proportions of coffee in cups is our invariant
- It does not allow us to have more milk than coffee in the first cup
- Indeed, otherwise we have more milk than coffee in both cups

More Coffee

Claim

The proportion of coffee in the first cup is always greater than in the second cup. That is, coffee in the first cup is stronger.

- Inequality between proportions of coffee in cups is our invariant
- It does not allow us to have more milk than coffee in the first cup
- Indeed, otherwise we have more milk than coffee in both cups
- But the total amount of coffee and milk is the same

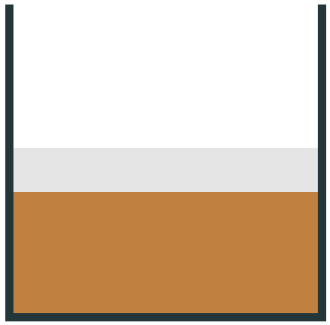
More Coffee

Claim

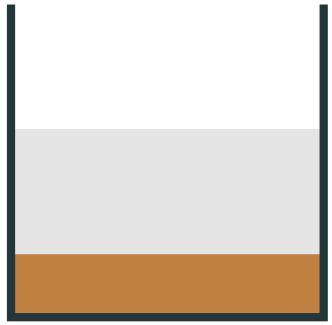
The proportion of coffee in the first cup is always greater than in the second cup. That is, coffee in the first cup is stronger.

Why the claim is true?

Cup 1



Cup 2

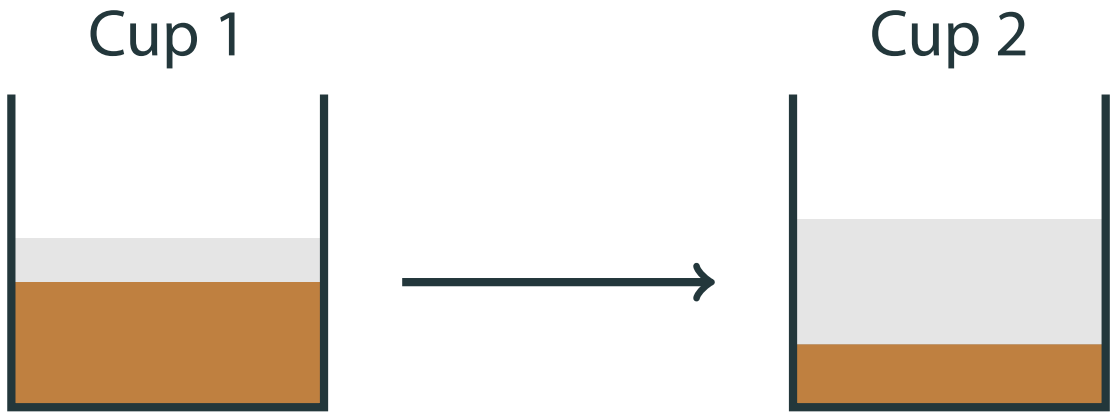


More Coffee

Claim

The proportion of coffee in the first cup is always greater than in the second cup. That is, coffee in the first cup is stronger.

Why the claim is true?

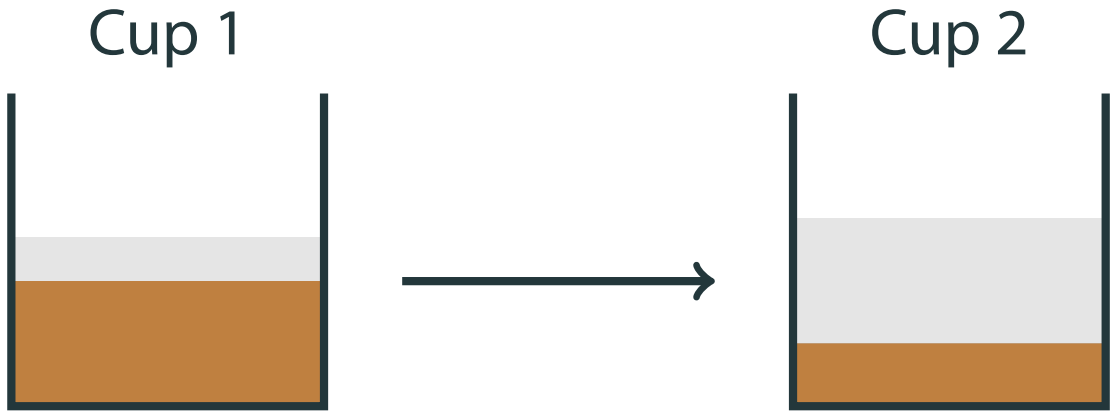


More Coffee

Claim

The proportion of coffee in the first cup is always greater than in the second cup. That is, coffee in the first cup is stronger.

Why the claim is true?



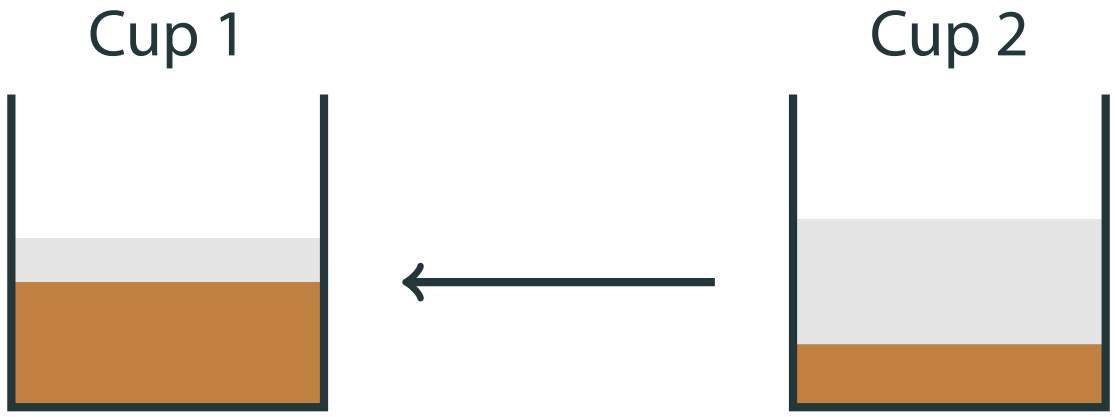
We “milk down” drink from Cup 1 in Cup 2

More Coffee

Claim

The proportion of coffee in the first cup is always greater than in the second cup. That is, coffee in the first cup is stronger.

Why the claim is true?

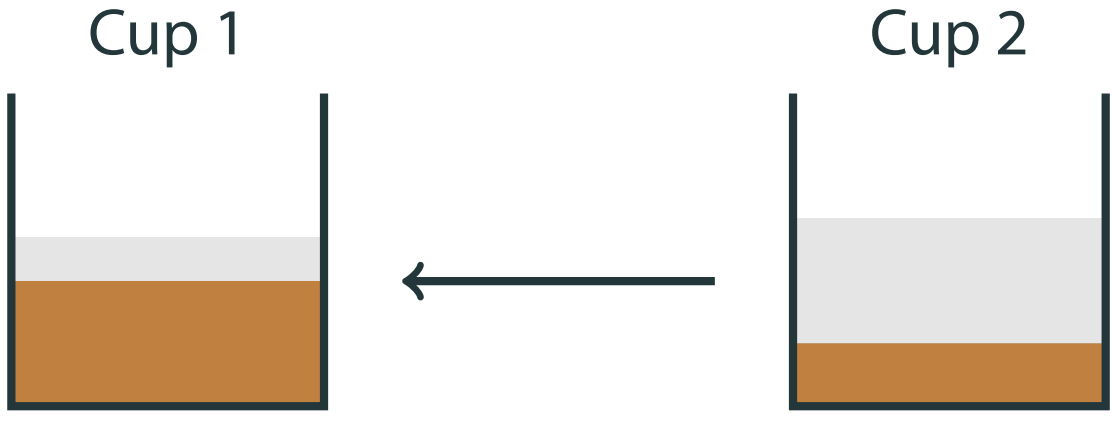


More Coffee

Claim

The proportion of coffee in the first cup is always greater than in the second cup. That is, coffee in the first cup is stronger.

Why the claim is true?



We “coffee down” drink from Cup 2 in Cup 1

Outline

Invariants

Coffee with milk

More Coffee

Debugging Problem

(Typical) Debugging

Problem

Bob is debugging his code. When he starts, he has only one bug. But once he fixes one bug, 3 new bugs appear. In several hours Bob fixed 15 bugs. How many pending bugs Bob has at this point?

(Typical) Debugging

Problem

Bob is debugging his code. When he starts, he has only one bug. But once he fixes one bug, 3 new bugs appear. In several hours Bob fixed 15 bugs. How many pending bugs Bob has at this point?

Let's see what's going on

(Typical) Debugging

Problem

Bob is debugging his code. When he starts, he has only one bug. But once he fixes one bug, 3 new bugs appear. In several hours Bob fixed 15 bugs. How many pending bugs Bob has at this point?

Let's see what's going on

Fixed	0	1				
Pending	1					

(Typical) Debugging

Problem

Bob is debugging his code. When he starts, he has only one bug. But once he fixes one bug, 3 new bugs appear. In several hours Bob fixed 15 bugs. How many pending bugs Bob has at this point?

Let's see what's going on

Fixed	0	1				
Pending	1	3				

(Typical) Debugging

Problem

Bob is debugging his code. When he starts, he has only one bug. But once he fixes one bug, 3 new bugs appear. In several hours Bob fixed 15 bugs. How many pending bugs Bob has at this point?

Let's see what's going on

Fixed	0	1	2			
Pending	1	3				

(Typical) Debugging

Problem

Bob is debugging his code. When he starts, he has only one bug. But once he fixes one bug, 3 new bugs appear. In several hours Bob fixed 15 bugs. How many pending bugs Bob has at this point?

Let's see what's going on

Fixed	0	1	2			
Pending	1	3	5			

(Typical) Debugging

Problem

Bob is debugging his code. When he starts, he has only one bug. But once he fixes one bug, 3 new bugs appear. In several hours Bob fixed 15 bugs. How many pending bugs Bob has at this point?

Let's see what's going on

Fixed	0	1	2	3		
Pending	1	3	5			

(Typical) Debugging

Problem

Bob is debugging his code. When he starts, he has only one bug. But once he fixes one bug, 3 new bugs appear. In several hours Bob fixed 15 bugs. How many pending bugs Bob has at this point?

Let's see what's going on

Fixed	0	1	2	3		
Pending	1	3	5	7		

(Typical) Debugging

Problem

Bob is debugging his code. When he starts, he has only one bug. But once he fixes one bug, 3 new bugs appear. In several hours Bob fixed 15 bugs. How many pending bugs Bob has at this point?

Let's see what's going on

Fixed	0	1	2	3	4	
Pending	1	3	5	7		

(Typical) Debugging

Problem

Bob is debugging his code. When he starts, he has only one bug. But once he fixes one bug, 3 new bugs appear. In several hours Bob fixed 15 bugs. How many pending bugs Bob has at this point?

Let's see what's going on

Fixed	0	1	2	3	4	
Pending	1	3	5	7	9	

(Typical) Debugging

Problem

Bob is debugging his code. When he starts, he has only one bug. But once he fixes one bug, 3 new bugs appear. In several hours Bob fixed 15 bugs. How many pending bugs Bob has at this point?

Let's see what's going on

Fixed	0	1	2	3	4	…
Pending	1	3	5	7	9	…

(Typical) Debugging

Fixed	0	1	2	3	4	…
Pending	1	3	5	7	9	…

- #Pending = 1 + 2×#Fixed

(Typical) Debugging

Fixed	0	1	2	3	4	…
Pending	1	3	5	7	9	…

- #Pending = 1 + 2×#Fixed
- This is our invariant!

(Typical) Debugging

Fixed	0	1	2	3	4	…
Pending	1	3	5	7	9	…

- #Pending = 1 + 2×#Fixed
- This is our invariant!
- When #Fixed=15 we have #Pending=31