Name: Flori Kusari

Date: 28/09/2023

ICS #3

Problem 3.1: Prove the following distributivity law for sets.

$$A \cap (B \cup C) = (A \cap B) \cup (A \cap C)$$

We use the theory of definitions of sets' intersections and unions.

Required Inclusions: 1. $A \cap (B \cup C) \subseteq (A \cap B) \cup (A \cap C)$

2. $(A \cap B) \cup (A \cap C) \subseteq A \cap (B \cup C)$

① Take an element "x" in $A \cap (B \cup C)$. By definition that element is both in A and in $(B \cup C)$. If the element "x" is in $B \cup C$, it means it is either in B or in C, by definition.

Case 1: For "x" in B: "x" is in both A and B so it is in $A \cap B$

Case 2: For "x" in C: "x" is in both A and C which is "x" in $A \cap C$.

If "x" is in either of $A \cap B$ or $A \cap C$, we conclude that "x" is the union of the sets which is $(A \cap B) \cup (A \cap C)$. Since we know one of them has to be true by definition, we can say that the conclusion is correct and the first inclusion has been proven true.

② Take an element "x" in $(A \cap B) \cup (A \cap C)$. By definition, the element is in either $(A \cap B)$ or in $(A \cap C)$. Lets say "x" in $(A \cap B)$ that means the "x" is in both A and B and it being in A also concludes it is also in $(B \cup C)$.

BRUNNEN

"x" in $(A\cap C)$ means it is in both sets. Since it is in C and A it is in $A\cap(B\cup C)$ also which, by definition, proves the statement which is the second inclusion: $(A\cap B)\cup(A\cap C)\subseteq A\cap(B\cup C)$.

Since both ① and ② have been proven we can say that it has been concluded that $A\cap(B\cup C)=(A\cap B)\cup(A\cap C)$ which proves the law of set distributity.

Problem 3.2: Prove or disprove the following two propositions.

a) $(A\cap B)\times(C\cap D)=(A\times C)\cap(B\times D)$

Take element "x" and element "y" in the proposition. "x" is in both A and B while "y" is in both C and D by following the proposition. (x,y) is in both $(A\times C)$ and $(B\times D)$ since "x" is in A, B while "y" is in C, D. Thus,

$$(x,y)\in(A\cap B)\times(C\cap D)\Longrightarrow(x,y)\in(A\times C)\cap(B\times D)$$

For the proof to be complete we must prove in both directions so:

For element "x" and element "y" in $(A\times C)\cap(B\times D)$, (x,y) is present in both $(A\times C)$ and $(B\times D)$. This implies that A and B are sets for "x" and C and D are sets for "y". Therefore:

$$(x,y)\in(A\times C)\cap(B\times D)\Longrightarrow(x,y)\in(A\cap B)\times(C\cap D)$$

Since both directions are correct we can conclude that:

$$(A\cap B)\times(C\cap D)=(A\times C)\cap(B\times D)$$

is true.

b) For the proposition: $(A \cup B) \times (C \cup D) = (A \times C) \cup (B \times D)$

we are going to try and prove it is false using a counter example

if we fail to prove it is false, then it must be true.

Take the sets: $A = \{1, 2\}$

$B = \{3, 4\}$

$C = \{a, b\}$

$D = \{c, d\}$

Now we calculate the propositions' sides and check if they are equal:

$$(A \cup B) \times (C \cup D) = (\{1, 2, 3, 4\}) \times (\{a, b, c, d\}) = \{(1,a), (1,b), (1,c), (1,d), (2,a), (2,b), (2,c), (2,d) \ldots (4,d)\}$$

$$(A \times C) \cup (B \times D) = (\{(1,a), (1,b), (2,a), (2,b)\}) \cup (\{(3,c), (3,d), (4,c), (4,d)\})$$

$$= \{(1,a), (1,b), (2,a), (2,b), (3,c), (3,d), (4,c), (4,d)\}$$

Since these two outputs are not equal the proposition:

b) $(A \cup B) \times (C \cup D) = (A \times C) \cup (B \times D)$

is False

Problem 3.3: For each of the following relations, determine whether they are reflexive, symmetric, or transitive. Provide a reasoning.

a)

$$R = \{(a, b) \mid a, b \in \mathbb{Z} \land |a - b| \leq 3\}$$

Let Reflexive be ①; Symmetry be ②; Transitivity be ③:

BRUNNEN

① The relation R is reflexive if (a,a) is in R for all "a" in a set. Therefore, for an integer "a" $|a-a| = 0$, which is ≤ 3 which leads to the relation being reflexive for all integers (a,a) in R relation.

② The relation R is symmetric for whenever (a,b) is in relation R, it's "symmetry" (b,a) must also be in R. If $|a-b| \leq 3$, then $|b-a| \leq 3$ since $|b-a| = |-(a-b)| = |a-b| \leq 3$. This leads to the relation R being symmetric as long as (a,b) and (b,a) are in R.

③ For trasitive lets test it using: $a = 1$

$b = 3$

$c = 5$

For (a,b): $|1-3| = 2 \leq 3$. So (a,b) is in R.

For (b,c): $|3-5| = 2 \leq 3$. So (a,c) is in R

For (a,c): $|1-5| = 4 \leq 3$. Which is not correct and (a,c) is not in R.

Therefore, the relation is **NOT** transitive.

a) is true for ①, ②, and False for ③

① For any "a" integer the (a mod 10) will give the last digit of said integer in relation R. Since the last integer of "a" will always be the same within 1 input, (a,a) is in R relation for all integers of a, thus, proving the relation reflexive.

② For R relation to be symmetric: If (a,b) in R, then $(a \bmod 10) = (b \bmod 10)$ which means the last digit of "a" and "b" must be the same. As long as this is true: (a,b) in R and (b,a) in R, then the relation is symmetric.

③ For transitive lets test it using:

a: 7

b: 17

c: 27

For (a,b): $(7 \bmod 10) = 7$ and $(17 \bmod 10) = 7$. Since $7 = 7$, (a,b) is in R.

For (b,c): $(17 \bmod 10) = 7$ and $(27 \bmod 10) = 7$. Since $7 = 7$, (b,c) is in R

For (a,c): $(7 \bmod 10) = 7$ and $(27 \bmod 10) = 7$. Since $7 = 7$, (a,c) is in R.

So the relation is transitive.

b) is true for 1, 2, and 3.

Problem 3.4:

a) zip :: [a] -> [b] -> [(a,b)]

The `zip` function is stated above and it takes two type variables "a" and "b" which are lists that are input into the `zip`. It pairs elements of the lists into tupples: (a, b). The number of variables it takes is set and it doesn't make sense to have less since two are needed and two are used. Since it is made to take one element from each list it will do exactly that for the two lists. It could be possible to use multiple `zip` if needed for the code but the maximum nr. of values per `zip` will not change.

b) Analyze the types of the following:

1. 2+3 : This is addition performed among two integers so the type is `Int`. Output = 5
2. 2+9`div`3 : The use of `div` as the integer divisor operator shows that it takes two Integers and produces an integer and adds it to 2. The output is 5 and the type is `Int`.
3. 2+9/3 : The 9/3 is a floating-point division and it gives a floating-point number which causes the result to be a floating-point number as well. The output is 5.0 which is a `Float` but in some other context it could be a `Double`.

BRUNNEN

④ 2 + sqrt 9 : The `sqrt` function makes this result a floating-point number so adding an integer to it will still remain as a floating-point number. The output is 5.0 which is a `Float` but under a different context it could be a `Double`

THE END