Exercises 4

Exercise 1 (Weakest Prediction)**.** Recall the definition of weakest precondition:

$$\mathrm{wp}(r, Q) = \{s \mid \forall s'.\ (s, s') \in r \to s' \in Q\}$$

1. Prove or disprove the following properties:
    - $\mathrm{wp}(r_1 \cup r_2, Q) = \mathrm{wp}(r_1, Q) \cup \mathrm{wp}(r_2, Q)$
    - $\mathrm{wp}(r_1 \cup r_2, Q) = \mathrm{wp}(r_1, Q) \cap \mathrm{wp}(r_2, Q)$
    - $\mathrm{wp}(r, Q_1 \cap Q_2) = \mathrm{wp}(r, Q_1) \cap \mathrm{wp}(r, Q_2)$
    - $\mathrm{wp}(r, Q_1 \cup Q_2) = \mathrm{wp}(r, Q_1) \cup \mathrm{wp}(r, Q_2)$

    For those that are wrong, do any of them hold if r is restricted to being functional, i.e. if r satisfies

    $$\forall x, y_1, y_2.\ ((x, y_1) \in r \land (x, y_2) \in r) \to (y_1 = y_2)$$

2. Let $r \subseteq S \times S$ and $Q \subseteq S$. Give an expression defining weakest precondition $\mathsf{wp}(r, Q)$ using operations of inverse of a relation, $(\_^{-1})$, set difference $(\setminus)$, and image of a relation under a set, $(\_[\_])$. Prove that your expression is correct by expanding the definitions of wp as well as of relational and set operations.

Exercise 2 (Programing With Integers)**.** Consider the following program:

```
case class Container(var x: INT, var y: INT):
  def fun: Unit = {
    require(x > 0 && y > 0)
    if x > y then
      x = x+y
      y = x-y
      x = x-y
    else
      y = y-x
  }.ensuring(2*old(this).x + old(this).y > 2*x + y &&
              x >= 0 && y >= 0)
end Container
```

- Compute $R(\text{fun})$ formally, by expressing all intermediate formulas corresponding to subprograms.
- Write the formula expressing the correctness of the ensuring clause. Is the formula valid when INT denotes mathematical integers with their usual operations (**type** INT = BigInt in Scala)?
- Is the the verification condition formula valid for machine integers,

$$Z_{2^{32}} = \{-2^{31}, \ldots, -1, 0, 1 \ldots, 2^{31} - 1\}$$

  and where operations are intepreted as the usual machine arithmetic (**type** INT = Int in Scala)?

Exercise 3 (Hoare Logic Proof)**.** Give a complete Hoare logic proof for the following program:

```
{n >= 0 && d > 0}
  q = 0
  r = n
  while ( r >= d ) {
    q = q + 1
    r = r - d
  }
{n == q * d + r && 0 <= r < d}
```

The proof should include step-by-step annotation for each line of the program, as in the example proof in the lecture.

Exercise 4 (Iterating a Relation)**.** Let $M = (S, I, r, A)$ be a transition system and $\bar{r} = \{(s, s') \mid (s, a, s') \in r\}$, as usual. Let $\Delta = \{(x, x) \mid x \in S\}$.

Let $\bar{r}^k$ denote the usual composition of relation $\bar{r}$ with itself k times.

Define the sequence of relations r_n, for all non-negative integers n, as follows:

- $r_0 = \Delta \cup \bar{r}$
- $r_{n+1} = r_n \circ r_n$

A) (2pt) Prove that $r_n \subseteq r_{n+1}$ for every n.

B) (3pt) Prove that for every n and every k where $0 \leq k \leq 2^n$ we have $\bar{r}^k \subseteq r_n$.

C) (2pt) Suppose that S is finite. Find a bound B as a function of $|S|$ such that

$$\mathsf{Reach}(M) \subseteq r_B[I]$$

Aim to find as small bound as possible.

Exercise 5 (Approximating Relations)**.** Consider a guarded command language whose meanings are binary relations on the set states U.

Let $E(a_1, \ldots, a_n)$ denote an expression built from some atomic relations $a_1, \ldots, a_n$, as well as diagonal relations

$$\Delta_P = \{(x, x) \mid x \in P\}$$

for various sets $P \subseteq U$. The expression E is built from these relations using union (to model non-deterministic choice, relation composition (to represent sequential composition) and transitive closure (to represent loops).

Let us call a relation $s \subseteq U \times U$ an *effect* if it is reflexive (R) and transitive (T).

A) (2pt) Prove that if s is an effect and $a_i \subseteq s$ for all $1 \leq i \leq n$, then

$$E(a_1, \ldots, a_n) \subseteq s$$

B) (2pt) Let $U = \mathbb{Z}^2$ denote pairs of integers, denoted by integer variables x, y. Let s be a specification relation given by the formula:

$$s = \{((x, y), (x', y')) \mid y \geq 0 \rightarrow (x' \leq x \land y' \geq 0)\}$$

Show that s is an effect.

C) (3pt) For the effect s in the previous point, prove that $\rho(p) \subseteq s$ where p is the following program (the initial values of variables can be arbitrary):

```
while (y ≥ 0) {
  x = x − y;
  if (x % 2 == 0)
    y = y / 2
  else
    y = 3*y + 1
}
```

Notation $\rho(p)$ denotes the relation corresponding to the program p.