Proof of total correctness

The following program is intended to divide two positive integers. Prove that it is partially correct w.r.t. the given specification according to the Hoare axiomatization (and the axioms of arithmetic, together with the natural deduction proof system) by choosing an appropriate invariant for the loop. Then prove it is totally correct by choosing an appropriate Floyd expression and showing that the modified Hoare axiom for while loops applies.

\{x \geq 0 \& y > 0\}
\text{q := 0;}
\text{r := x;}
\text{while r \geq y do}
  \begin{align*}
    \text{r := r - y;}
    \text{q := q + 1;}
  \end{align*}
\text{end;}
\{x = q \times y + r \& 0 \leq r < y\}

Points to note:
- The specification could have been written as \(q = x / y\), but the one given is simpler and is in any case equivalent.
- There are many invariants but only one which allows the proof to succeed. Be careful to complete the proof properly with your invariant.
- Where there are conditional proofs involving arithmetic expressions (the “glue”) you must justify the truth of the conditional in a discussion of cases rather than just stating it as “obvious” (even though it may be). If you want to write a complete proof go ahead, but it makes the whole thing very long indeed.
- To prove total correctness it is not necessary to repeat the partial correctness proof; you can eliminate the important part of the invariant.

You may get help from other people in the class. Please indicate on your answer sheet whom you collaborated with. Any indication of unauthorized copying may be penalized with a zero for the assignment.

Due date: December 13th.