Exercise 1: (Optimal) Register allocation for basic blocks

Consider the following basic block,

\[
\begin{align*}
  t &= 0; \\
  x &= t; \\
  y &= 1; \\
  z &= x + y; \\
  x &= x + 1; \\
  q &= x + y; \\
  y &= q + z; \\
  t &= t + 1; \\
  x &= x + 1; \\
  r &= y + t; \\
  ret &= x + r;
\end{align*}
\]

and the corresponding CFG:

- Compute the true liveness information for each program point. (The variable \textit{ret} is the only live variable at the end of this block.)
- Construct the interference graph.
- Make use of the Greedy Heuristics in order to provide a coloring for the graph.
- Construct the interval graph. (Note that you should split live ranges whenever possible, even when it is obvious that it will not improve register allocation.)
e) Provide an optimal coloring for the graph.

Exercise 2: Static Single Assignment (SSA)
Bring the following program into SSA form, i.e.

a) compute the information of reachability for each node and perform the first transformation—which might introduce edges with parallel assignments,

b) compute the information of reachability for each node again and perform the second transformation—which relabels the variables.

Assume that all variables are live.