

6th practice

- Integers are stored in a sequential input file sorted in ascending order. Count how many times each number occurs in the file and write the (number, count) records into a sequential output file.

Specification:

$A = (x:\text{infile}(\mathbb{Z}), y:\text{outfile}(\text{Stat}))$
 $\text{Stat} = \text{rec}(\text{num}:\mathbb{Z}, \text{count}:\mathbb{N})$
 $\text{Pre} = (x = x_0 \wedge x \nearrow)$
 $(x \nearrow \text{ denotes that } x \text{ is sorted in an ascending order})$

New Specification:

$A = (t:\text{enor}(\text{Stat}), y:\text{outfile}(\text{Stat}))$
 $\text{Pre} = (t = t_0)$
 $\text{Post} = (y = t_0) = (y = \bigoplus_{e \in t_0} \langle e \rangle)$

Summation (copying)

$f(e) \sim \langle e \rangle$
 $s \sim y$
 $H, +, 0 \sim \text{Stat}^*, \bigoplus, \langle \rangle$

Idea:

Enumerate the (number, count) records and copy them into the output file.

Algorithm:

$y := \langle \rangle$
$t.\text{first}()$
$\neg t.\text{end}()$
$y : \text{write}(t.\text{current}())$
$t.\text{next}()$

Enumerator:

$t:\text{enor}(\text{Stat}) \quad \text{Stat} = \text{rec}(\text{num}:\mathbb{Z}, \text{count}:\mathbb{N})$

Stat^*	$\text{first}()$	$\text{next}()$	$\text{current}() : \text{Stat}$	$\text{end}() : \mathbb{L}$
$x : \text{infile}(\mathbb{Z})$ $dx : \mathbb{Z}$ $sx : \text{Status}$ $\text{curr} : \text{Stat}$ $\text{end} : \mathbb{L}$	$sx, dx, x : \text{read next}()$	see below	return curr	return end

next() method

$S = (x:\text{infile}(\mathbb{Z}), dx:\mathbb{Z}, sx:\text{Status}, \text{curr}:\text{Stat}, \text{end}:\mathbb{L})$

$\text{Pre} = (x = x' \wedge x \nearrow \wedge dx = dx' \wedge sx = sx')$

$dx = \text{curr.num}$

$\text{Post} = (\text{end} = (sx' = \text{abnorm}) \wedge (\neg \text{end} \rightarrow \text{curr.num} = dx' \wedge (\text{curr.count}, (sx, dx, x)) = \sum_{dx \in (dx', x')} 1))$

Note: Summation has two results: the count (curr.count); and the current state of the enumerator, which is the value of the variables sx, dx, x after the next() method has finished.

Summation (counting)

$t:\text{enor}(E) \sim x:\text{infile}(\mathbb{Z}) (sx, dx, x : \text{read})$
 without first(),
 as long as: $dx = \text{curr.num}$

$f(e) \sim 1$
 $s \sim \text{curr.count}$
 $H, +, 0 \sim \mathbb{N}, +, 0$

$\text{end} := sx = \text{abnorm}$	
$\neg \text{end}$	
$\text{curr.num}, \text{curr.count} := dx, 0$	-
$sx = \text{norm} \wedge dx = \text{curr.num}$	
$\text{curr.count} := \text{curr.count} + 1$	
$sx, dx, x : \text{read}$	