MODULE *knapsack*

EXTENDS *TLC*, *Integers*, *Sequences*

We have a knapsack of volume N and a set of items. Each item has a value and a size. You can fit any number of each item in the knapsack as long as the sum of them all is less than the capacity of the sack. What'*s* the most valuable knapsack you can make?

VARIABLES *pc*, *chosen*

$Capacity \triangleq 4$

$Items \triangleq \{\text{“a”}, \text{“b”}, \text{“c”}\}$

$ItemParams \triangleq [size : 1 \,..\, 4, value : 0 \,..\, 5]$

$ItemSets \triangleq [Items \to ItemParams]$

$Knapsacks \triangleq [Items \to 0 \,..\, 4]$

$HardcodedItemSet \triangleq [$
$\quad a \mapsto [size \mapsto 1, value \mapsto 1],$
$\quad b \mapsto [size \mapsto 2, value \mapsto 3],$
$\quad c \mapsto [size \mapsto 3, value \mapsto 1]$
$]$

$ReduceSet(op(\_, \_), set, acc) \triangleq$
Shamelessly stolen from *PT* library
LET $f[s \in \text{SUBSET } set] \triangleq$
IF $s = \{\}$ THEN acc
ELSE LET $x \triangleq \text{CHOOSE } x \in s : \text{TRUE}$
IN $op(x, f[s \setminus \{x\}])$
IN $f[set]$

$KnapsackSize(sack, itemset) \triangleq$
LET $size\_for(item) \triangleq itemset[item].size * sack[item]$
IN $ReduceSet(\text{LAMBDA } item, acc : size\_for(item) + acc, Items, 0)$

$KnapsackValue(sack, itemset) \triangleq$
LET $value\_for(item) \triangleq itemset[item].value * sack[item]$
IN $ReduceSet(\text{LAMBDA } item, acc : value\_for(item) + acc, Items, 0)$

$ValidKnapsacks(itemset) \triangleq$
$\{sack \in [Items \to 0 \,..\, 4] : KnapsackSize(sack, itemset) \leq Capacity\}$

$BestKnapsackSlow(itemset) \triangleq$

LET $all \triangleq ValidKnapsacks(itemset)$
$val(k) \triangleq KnapsackValue(k,\ itemset)$

IN CHOOSE $all\_the\_best \in$ SUBSET all : there exists a subset of all knapsacks such that
$\land\ all\_the\_best \neq \{\}$ is not empty
$\land\ \forall\ good \in all\_the\_best$:
$\land\ \forall\ best \in all\_the\_best : val(best) = val(good)$ every knapsack has the same value
$\land\ \forall\ not\_best \in (all \setminus all\_the\_best) : val(good) > val(not\_best)$ it's value is greater than all other kn

$BestKnapsack(itemset) \triangleq$

Select the knapsack with the highest value, then select all *Knapsacks* with that same value

LET $all \triangleq ValidKnapsacks(itemset)$
$val(k) \triangleq KnapsackValue(k,\ itemset)$
$best \triangleq$ CHOOSE $x \in all : \forall\ y \in (all \setminus \{x\}) : val(x) \geq val(y)$
IN $\{k \in all : val(k) = val(best)\}$

$Init \triangleq$
$\land\ pc =$ "init"
$\land\ chosen = \{\}$

$PickBest \triangleq$
$\land\ pc =$ "init"
$\land\ chosen' = \{BestKnapsack(it) : it \in ItemSets\}$
$\land\ pc' =$ "done"

$Done \triangleq$
$\land\ pc =$ "done"
$\land\ PrintT(\langle$"best knapsack", $chosen\rangle)$
$\land$ UNCHANGED $\langle pc,\ chosen\rangle$

$Next \triangleq PickBest \lor Done$
