COLLEGE OF COMPUTING AND INFORMATION SCIENCES
SCHOOL OF COMPUTING AND INFORMATICS TECHNOLOGY
BACHELOR OF SCIENCE IN COMPUTER SCIENCE
GROUP COURSE WORK
COURSE NAME: FORMAL METHODS
BY : Mr.BITWIRE ALBERT
COURSE CODE :BSE 2105
GROUP 2 DAY

No.	**Student Name**	**RegNo**	**Signature**
1	**OBUR JOSEPH DUNATO**	17/U/9439/PS	
2	**MUSINGUZI JOSHUA**	17/U/6672/PS	
3	**SSENTONGO MARTIN L**	17/U/20558/PS	
4			

Question 1

(a)State Space Schema for WeatherMap

WeatherMap

$location : \mathbb{P}\, REGION$
$weather : REGION ⇸ TEMPERATURE$

$\operatorname{dom} weather = location$

(b) This schema updates the WeatherMap database

Update

$\Delta\, WeatherMap$
$reg? : REGION$
$temp? : TEMPERATURE$

$reg? \in location$
$weather' = weather \oplus \{reg? \mapsto temp?\}$

(c) A LookUp Schema that searches for the temperature of a particular region

LookUp

$\Xi\, WeatherMap$
$reg? : REGION$
$temp! : TEMPERATURE$

$reg? \in location$
$temp! = weather(\mathrm{reg?})$

(d) An Init-Map schema that shows the initial state

InitMap

$WeatherMap$

$weather' = \emptyset$

Question 2

Below are the basic types required for the vending machine
$[GOOD, reportortortortORT, GOOD]$

COIN==$\{100, 200, 500, 1000\}$
$GOOD == \{Mangojuice, Passionfruit_Juice, PineApple_Juice, Orange_Juice, Lemon_Juice, Cocktail_juice\}$
$\mathbb{N}$==$\{n : \mathbb{Z} \mid n \geq 0\}$
sum : bag COIN $\nrightarrow$ $\mathbb{N}$

reportortortortORT ::= Some goods are not priced | Item already priced | Item not in stock | More money is requred | You haave some balance | Operation is unsuccessful | Item out of stock

VENDING MACHINE

(i) The state space schema

VendingMachine

$Inventory : \text{bag}\ GOOD$
$Cost : GOOD \nrightarrow \mathbb{N}$
$float : \text{bag}\ COIN$
$accept : \mathbb{P}\ COIN$
$entered : \text{bag}\ COIN$

$\text{dom}\ inventory \subseteq price$
$\text{dom}\ entered \subseteq accepted$
$\text{dom}\ float \subseteq accepted$

(ii) Initialising the machine

InitVendingMachine

$\Delta\ VendingMachine$

$inventory' = [\![ ]\!]$
$price' = \emptyset$
$float' = [\![ ]\!]$
$accept' = [\![ ]\!]$
$entered' = [\![ ]\!]$

(iii) A schema used to give price to a product.

$$
\begin{array}{|l}
\hline
\mathit{Pricing} \\
\hline
\Xi \, \mathit{VendingMachine} \\
\mathit{item}? : \mathit{GOOD} \\
\mathit{cost}? : \mathbb{N} \\
\hline
\mathit{item}? \mapsto \mathit{price}? \notin \mathit{cost} \\
\mathit{price}' = \mathit{price} \oplus \{\mathit{item}? \mapsto \mathit{cost}?\} \\
\mathit{invetory}' = \mathit{invetory} \\
\hline
\end{array}
$$

(iv) AcceptCoin schema that adds a coin to the set of accepted coins.

$$
\begin{array}{|l}
\hline
\mathit{AcceptCoin} \\
\hline
\Delta \, \mathit{VendingMachine} \\
\mathit{newCoin}? : \mathit{COIN} \\
\hline
\mathit{newCoin}? \notin \mathit{accept} \\
\mathit{float}' = \mathit{float} \\
\mathit{entered}' = \mathit{entered} \\
\mathit{accept}' = \mathit{accept} \cup \{\mathit{newCoin}?\} \\
\hline
\end{array}
$$

(v) Schema specifying what happens when some one buys an item from the machine

$$
\begin{array}{|l}
\hline
\mathit{Buy} \\
\hline
\Delta \, \mathit{VendingMachine} \\
\mathit{item}? : \mathit{GOOD} \\
\hline
\mathit{item}? \text{ in } \mathit{inventory} \\
\mathit{inventory}' = \mathit{inventory} \uplus \llbracket \mathit{item}? \rrbracket \\
\mathit{price} = \mathit{price}' \\
\hline
\end{array}
$$

(vi) The Schema specifying what happens when the Item someone wants to buy is not in the machine

$$
\begin{array}{|l}
\hline
\mathit{NotInMachine} \\
\hline
\Xi \, \mathit{VendingMachine} \\
\mathit{item}? : \mathit{GOOD} \\
\mathit{reportortortort}! : \mathit{reportortortortORT} \\
\hline
(\mathrm{dom}\, \mathit{inventory}' \subseteq \mathit{price}') \\
\mathit{reportortortort}! = \textit{”Item is not in stock”} \\
\hline
\end{array}
$$

$$RobustPriceUpdate \mathrel{\widehat=} (priceUpdate \land success) \lor NotInSystem$$

(vii)The schema for the reportortortortort when the pricing of a good is successful

$$\begin{array}{|l}
\multicolumn{1}{l}{\underline{\quad Success \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad}} \\
reportortortort! : reportortortortORT \\
\hline
reportortortort! = \text{"}Good\ successfully\ priced\text{"} \\
\hline
\end{array}$$

The schema for Error

$$\begin{array}{|l}
\multicolumn{1}{l}{\underline{\quad Error \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad}} \\
reportortortort! : reportortortortORT \\
\hline
reportortortort! = \text{"}Operation\ is\ unsuccessful\text{"} \\
\hline
\end{array}$$

$$RobustSell \mathrel{\widehat=} (Buy \land success) \lor NotInStock \lor LessMoney \lor change \lor error$$

(viii)

- Not in stock

$$\begin{array}{|l}
\multicolumn{1}{l}{\underline{\quad NoStock \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad}} \\
\Xi VendingMachine \\
item? : GOOD \\
reportortortort! : reportortortortORT \\
\hline
(item?\ in\ stock) \\
reportortortort! = \text{``}Item\ out\ of\ stock\text{''} \\
\hline
\end{array}$$

- Less money

$$\begin{array}{|l}
\multicolumn{1}{l}{\underline{\quad LessMoney \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad}} \\
\Xi VendingMachine \\
In? : \text{bag}\ COIN; \\
item? : GOOD \\
reportortortort! : reportortortortORT \\
\hline
sum(in?) < cost(item?) \\
reportortortort! = \text{``}More\ Money\ Required\text{''} \\
\hline
\end{array}$$

- Already priced

$$
\begin{array}{|l}
\multicolumn{1}{l}{\mathit{AlreadyPriced}} \\
\hline
\Xi\, \mathit{VendingMachine} \\
\mathit{item}? : \mathit{GOOD} \\
\mathit{cost}? : \mathbb{N} \\
\mathit{reportortortort}! : \mathit{reportortortortORT} \\
\hline
\mathit{item}? \mapsto \mathit{cost}? \in \mathit{price} \\
\mathit{ortortort}! = \text{''}\mathit{Item\ already\ priced}\text{''} \\
\hline
\end{array}
$$

- What if i want to restock

$$
\begin{array}{|l}
\multicolumn{1}{l}{\mathit{Restock}} \\
\hline
\Delta\, \mathit{VendingMachine} \\
\mathit{newStock}? : \mathrm{bag}\ \mathit{GOOD} \\
\hline
\mathit{inventory}' = \mathit{inventory} \uplus \mathit{newStock}? \\
\mathit{price} = \mathit{price}' \\
\mathit{float} = \mathit{float}' \\
\hline
\end{array}
$$

- What if the good is not priced

$$
\begin{array}{|l}
\multicolumn{1}{l}{\mathit{GoodNotPriced}} \\
\hline
\Xi\, \mathit{VendingMachine} \\
\mathit{newStock}? : \mathrm{bag}\ \mathit{GOOD} \\
\mathit{reportortort}! : \mathit{reportortortORT} \\
\hline
\neg(\mathrm{dom}\, \mathit{newStock}? \subseteq \mathrm{dom}\, \mathit{price}) \\
\mathit{reportortort}! = \text{``}\mathit{Some\ Goods\ Not\ Priced}\text{''} \\
\hline
\end{array}
$$

- Give Change

$$
\begin{array}{|l}
\textit{Change} \\
\hline
\Delta \mathit{VendingMachine} \\
\mathit{cost}? : \mathbb{N} \\
\hline
\mathit{sumBag}(\mathit{entered}) \geq \mathit{cost}? \\
\mathit{sumBag}(\mathit{entered}') = \mathit{sumBag}(\mathit{entered}) - \mathit{cost}? \\
\mathit{float} \uplus \mathit{entered} = \mathit{float}' \uplus \mathit{entered}' \\
\mathit{accept}' = \mathit{accept} \\
\mathit{price}' = \mathit{price} \\
\mathit{reportortort}! = \textit{"You have some balance"} \\
\hline
\end{array}
$$

- What if i want to know number of a Good in the machine

$$
\begin{array}{|l}
\textit{CountGood} \\
\hline
\Xi \mathit{VendingMachine} \\
\mathit{item}? : \mathit{GOOD} \\
\mathit{amount}! : \mathbb{N} \\
\hline
\mathit{item}? \in \operatorname{dom} \mathit{price} \\
\mathit{amount}! = \mathit{inventory} \sharp \mathit{item}? \\
\hline
\end{array}
$$

- Incase i want to find the price of a product

$$
\begin{array}{|l}
\textit{FindPrice} \\
\hline
\Xi \mathit{VendingMachine} \\
\mathit{item}? : \mathit{GOOD} \\
\mathit{cost}! : \mathbb{N} \\
\hline
\mathit{item}? \in \operatorname{dom} \mathit{price} \\
\mathit{cost}! = \mathit{price}(\mathit{item}?) \\
\hline
\end{array}
$$