Formal Specification and Verification of Programs

2nd Assignment Solutions
Mohammad Hossein Khoshechin - 99210164
Group 2

۲۰ دی ۱۳۹۹

تمرین ۱ : بازی X-O

بازی X-O یک بازی دو نفره است که بر روی یک جدول انجام می شود . یک بازیکن به طور قردادی برای علامت زدن خانه های جدول از نماد X استفاده می کند و بازیکن دیگر از نماد O استفاده می کند.هر بازیکن زمانی که نوبتش رسید می تواند تنها در یکی از خانه های خالی جدول ، نمادش را قرار دهد . قرار داد می کنیم که در زمان شروع بازی ، نوبت متعلق به بازیکن X می باشد . هر بازیکن بعد از اینکه در نوبت خود یک خانه را علامت زد ، نوبت به بازیکن دیگر داده می شود. هر زمان که سه خانه از جدول به شکل سطری ، ستونی و یا قطری ، تنها با استفاده از یک نماد پر شود آنگاه بازیکنی که با آن نماد بازی میکرده ، برنده است . زمانی که تمام خانه های جدول پر شود اما هیچ یک از سطرها ، ستون ها و قطرها ، تنها با یک نماد پر نشده باشد آنگاه بازی مساوی می شود.

$$Index == \{a : \mathbb{N} \mid a < 3\}$$
$$Cell == Index \times Index$$
$$State == empty \mid x \mid o$$
$$Turn == x \mid o$$
$$Message == Cell\_Filled \mid Action\_Faild \mid Xwin \mid Owin \mid noWinner$$

GameTable

$$table : Cell \rightarrow State$$
$$pturn : Turn$$

GameTableInit

$$GameTable'$$

$$\forall c : Cell \bullet table'\ c = empty$$
$$pturn' = x$$

تا به اینجا state را تعریف کردیم که دارای یک جدول است. هر خانه از جدول هم به یک وضعیت مپ شده است در حالت initial هم تمام خانه ها را empty در نظر میگیرم و نوبت را به بازیکن x میدهیم.

حال دو تا Schema Operator مینویسیم برای بازی کردن بازکن x و بازیکن o . در هر کدام یک متغیر Cell به عنوان ورودی میگیریم و یک پیام output هم به عنوان خروجی قرار میدهیم.

Oplay

$$
\begin{aligned}
&\Delta GameTable \\
&location? : Cell \\
&output! : Message \\
\hline
&pturn = o \\
&table\ location? = empty \\
&table' = table \oplus \{\, location? \mapsto o \,\} \\
&output! = Cell\_Filled \\
&pturn' = x
\end{aligned}
$$

Xplay

$$
\begin{aligned}
&\Delta GameTable \\
&location? : Cell \\
&output! : Message \\
\hline
&pturn = x \\
&table\ location? = empty \\
&table' = table \oplus \{\, location? \mapsto x \,\} \\
&output! = Cell\_Filled \\
&pturn' = o
\end{aligned}
$$

حال باید این نکته را در نظر بگیریم که ممکن است زمانی که یک بازیکن میخواهد خانه ای از جدول را پر کند ، آن خانه خالی نباشد یا نوبتش نباشد. پس باید یک Schema Operator هم برای هر دو بازی کن لحاظ کنیم .

OplayFaild

$\Xi GameTable$
$location? : Cell$
$output! : Message$

$pturn = x$
$table\ location? \neq empty$
$output! = Action\_Faild$

XplayFaild

$\Xi GameTable$
$location? : Cell$
$output! : Message$

$pturn = o$
$table\ location? \neq empty$
$output! = Action\_Faild$

حال با disjunction کردن Schema Operator ها ، Schema Operator مربوط به حرکت هر دو بازیکن را تعریف می کنیم :

$$MoveX == Xplay \lor XplayFaild$$
$$MoveO == Oplay \lor OplayFaild$$

حال با نوشتن چند Schema Operator، شرط خاتمه بازی را تعریف میکنیم . برای این کار باید سه تا Schema Operator بنویسیم یکی باری برد بازیکن x ، یکی برای برد بازیکن o و دیگری برای حالت تساوی.

Xwin

$\Xi GameTable$

$output! : Message$

$\exists\, i_1, i_2, i_3, j_1, j_2, j_3 : Index \bullet$

$$
\begin{aligned}
&((i_1 = i_2) \land (i_2 = i_3) \land (j_2 = j_1 + 1) \land (j_3 = j_2 + 1) \land \\
&\qquad (table(\{ (i_1, j_1), (i_2, j_2), (i_3, j_3) \}) = \{ x \})) \lor \\
&((j_1 = j_2) \land (j_2 = j_3) \land (i_2 = i_1 + 1) \land (i_3 = i_2 + 1) \land \\
&\qquad (table(\{ (i_1, j_1), (i_2, j_2), (i_3, j_3) \}) = \{ x \})) \lor \\
&((i_1, j_1) = (0, 0) \land (i_2, j_2) = (1, 1) \land (i_3, j_3) = (2, 2) \land \\
&\qquad (table(\{ (i_1, j_1), (i_2, j_2), (i_3, j_3) \}) = \{ x \})) \lor \\
&((i_1, j_1) = (0, 2) \land (i_2, j_2) = (1, 1) \land (i_3, j_3) = (0, 2) \land \\
&\qquad (table(\{ (i_1, j_1), (i_2, j_2), (i_3, j_3) \}) = \{ x \}))
\end{aligned}
$$

$output! = Xwin$

Owin

$\Xi GameTable$

$output! : Message$

$\exists\, i_1, i_2, i_3, j_1, j_2, j_3 : Index \bullet$

$$
\begin{aligned}
&((i_1 = i_2) \land (i_2 = i_3) \land (j_2 = j_1 + 1) \land (j_3 = j_2 + 1) \land \\
&\qquad (table(\{ (i_1, j_1), (i_2, j_2), (i_3, j_3) \}) = \{ o \})) \lor \\
&((j_1 = j_2) \land (j_2 = j_3) \land (i_2 = i_1 + 1) \land (i_3 = i_2 + 1) \land \\
&\qquad (table(\{ (i_1, j_1), (i_2, j_2), (i_3, j_3) \}) = \{ o \})) \lor \\
&((i_1, j_1) = (0, 0) \land (i_2, j_2) = (1, 1) \land (i_3, j_3) = (2, 2) \land \\
&\qquad (table(\{ (i_1, j_1), (i_2, j_2), (i_3, j_3) \}) = \{ o \})) \lor \\
&((i_1, j_1) = (0, 2) \land (i_2, j_2) = (1, 1) \land (i_3, j_3) = (0, 2) \land \\
&\qquad (table(\{ (i_1, j_1), (i_2, j_2), (i_3, j_3) \}) = \{ o \}))
\end{aligned}
$$

$output! = Owin$

WithDraw

$\Xi GameTable$
$output! : Message$

$\forall c : Cell \bullet table\ c \neq empty$
$\forall i_1, i_2, i_3, j_1, j_2, j_3 : Index \bullet$
$((i_1 = i_2) \land (i_2 = i_3) \land (j_2 = j_1 + 1) \land (j_3 = j_2 + 1)) \Rightarrow$
$(\# table(\{ (i_1, j_1), (i_2, j_2), (i_3, j_3) \}) \neq 1)) \land$
$((j_1 = j_2) \land (j_2 = j_3) \land (i_2 = i_1 + 1) \land (i_3 = i_2 + 1)) \Rightarrow$
$(\# table(\{ (i_1, j_1), (i_2, j_2), (i_3, j_3) \}) \neq 1)) \land$
$((i_1, j_1) = (0, 0) \land (i_2, j_2) = (1, 1) \land (i_3, j_3) = (2, 2)) \Rightarrow$
$(\# table(\{ (i_1, j_1), (i_2, j_2), (i_3, j_3) \}) \neq 1)) \land$
$((i_1, j_1) = (0, 2) \land (i_2, j_2) = (1, 1) \land (i_3, j_3) = (0, 2)) \Rightarrow$
$(\# table(\{ (i_1, j_1), (i_2, j_2), (i_3, j_3) \}) \neq 1))$
$output! = noWinner$

$End == Xwin \lor Owin \lor WithDraw$

$XOGame == MoveX \lor MoveO \lor End$

تمرین ۲ : بازی pac-man

بازی پکمن به این صورت است که در یک جدول انجام میشود . بعضی از خانه های جدول دیوار wall هستند و بعضی دیگر مسیر path. Agent های بازی به شکل پکمن ، روح ها و دانه ها هستند . ما به عنوان بازیکن تنها هدایت پکمن را بر عهده داریم . طبیعتا پکمن تنها در خانه هایی که مسیر هستند می تواند حرکت کند و نمی تواند وارد خانه هایی شود که دیوار هستند . پکمن زمانی که در یک خانه وارد میشود اگر در آن خانه دانه باشد آن را می خورد و اگر روح باشد کشته می شود و بازی تمام می شود . زمانی برنده می شویم که پکمن تمام دانه های موجود در جدول را بخورد. دانه ها تنها در خانه هایی که دیوار نیستند قرار داده می شوند . حرکت روح ها در این سناریو به طور تصادفی است و هدفمند نیست . روح ها نیز مانند پکمن نمی توانند وارد خانه هایی شوند که دیوار است . دیوار ها باید طوری در جدول قرار گیرند که بین هر دو خانه ای که در جدول دیوار نیستند(path هستند) ، مسیری از خانه های path وجود داشته باشد.

$Cell == \mathbb{N} \times \mathbb{N}$
$State == wall \mid path$
$Message == win \mid defeated$
$Move == UP \mid DOWN \mid LEFT \mid RIGHT$

$$
\begin{array}{|l}
\textit{GameTable} \\
\hline
table : Cell \rightarrow State \\
pacman : Cell \\
ghosts : \mathbb{P}\ Cell \\
seeds : \mathbb{P}\ Cell \\
length : \mathbb{N} \\
width : \mathbb{N} \\
\hline
\forall i, j : \mathbb{N} \mid (i, j) \in dom\, table \bullet i < width \land j < length \\
\exists c_1, c_2, c_3 : Cell \bullet (c_1 \neq c_2) \land (c_2 \neq c_3) \land (table(\{ c_1, c_2, c_3 \}) = \{ path \}) \land \\
\qquad \{ c_1, c_2, c_3 \} \subset dom\ table \forall c : Cell \mid c \in seeds \bullet table\ c = path \\
\# gosts \geq 1 \\
\# seeds \geq 1 \\
table\, pacman = path \\
pacman \in dom\ table \\
seeds \subset dom\ table \\
ghosts \subset dom\ table \\
table(ghosts) = \{ path \} \\
table(seeds) = \{ path \} \\
\forall c_1, c_2 : Cell \mid c_1 \in dom\ table \land c_2 \in dom\ table \land c_1 \neq c_2 \bullet (table(\{ c_1, c_2 \}) = \{ path \}) \Rightarrow \\
\quad (\exists road : Seq\ Cell \bullet head\ road = c_1 \land (\forall x : \mathbb{N} \mid x \geq 1 \land x < \# road \bullet \\
\qquad table(\{ road\ x, road\ x + 1 \}) = \{ path \} \land \\
\qquad (\exists i_1, i_2, j_1, j_2 : \mathbb{N} \mid (i_1, j_1) = road\ x \land (i_2, j_2) = road\, x + 1 \bullet (i_1 = i_2 + 1 \land j_1 = j_2) \lor \\
\qquad\quad (j_1 = j_2 + \land\ i_1 = i_2))) \land \\
\quad road\ \#road = c_2) \\
\hline
\end{array}
$$

اندازه جدول توسط بازیکن قبل از شروع بازی مشخص می شود و اندازه جدول حداقل 3*3 باید باشد. تعداد روح ها و دانه ها و جایگاهشان همراه با جایگاه پکمن در ابتدای بازی به شکل تصادفی انتخاب می شوند .

$$
\begin{array}{|l}
\textit{GameTableInit} \\
\hline
GameTable' \\
sampleLength? : \mathbb{N} \\
sampleWidth? : \mathbb{N} \\
\hline
pacman' \notin gosts' \\
table' = sampleTable \\
pacman' = samplePacman \\
ghosts' = sampleGhosts \\
seeds' = sampleSeeds \\
length' = sampleLength? \\
width' = sampleWidth? \\
sampleWidth \geq 3 \\
sampleLength \geq 3 \\
pacman' \neq \varnothing \\
\hline
\end{array}
$$

$sampleTable : Cell \rightarrow State$

$samplePacman : Cell$

$sampleGhosts : \mathbb{P}\ Cell$

$sampleSeeds : \mathbb{P}\ Cell$

در این قسمت حرکت پکمن را توصیف کرده ایم.

PMoveUp

$\Delta GameTable$
$i : \mathbb{N}$
$j : \mathbb{N}$
$direction? : Move$

$direction? = UP$
$table' = table$
$(i, j) = pacman$
$table(i-1, j) \neq wallpacman' = (i-1, j)$
$ghosts' = ghosts$
$seeds' = seeds$
$length' = length$
$width' = width$

PMoveDown

$\Delta GameTable$
$i : \mathbb{N}$
$j : \mathbb{N} direction? : Move$

$direction? = DOWN$
$table' = table$
$(i, j) = pacman$
$table(i+1, j) \neq wallpacman' = (i+1, j)$
$ghosts' = ghosts$
$seeds' = seeds$
$length' = length$
$width' = width$

PMoveRight

$\Delta GameTable$
$i : \mathbb{N}$
$j : \mathbb{N} direction? : Move$

$direction? = Right$
$table' = table$
$(i, j) = pacman$
$table(i, j + 1) \neq wall pacman' = (i, j + 1)$
$ghosts' = ghosts$
$seeds' = seeds$
$length' = length$
$width' = width$

PMoveLeft

$\Delta GameTable$
$i : \mathbb{N}$
$j : \mathbb{N}$
$direction? : Move$

$direction? = LEFT$
$table' = table$
$(i, j) = pacman$
$table(i, j - 1) \neq wall pacman' = (i, j - 1)$
$ghosts' = ghosts$
$seeds' = seeds$
$length' = length$
$width' = width$

$MovePacman == PMoveUp \lor PMoveDown \lor PMoveLeft \lor PMoveRight$

در این قسمت عملات خوردن دانه توسط پکمن را توصیف کرده ایم.

EatSeed

$\Delta GameTable$

$table' = table$
$pacman \in seeds$
$seeds' = seeds \backslash pacman$
$ghosts' = ghosts$
$length' = length$
$width' = width$

در این قسمت حرکت روح ها را که قرار است تصادفی و غیر هوشمندانه باشد را توصیف کرده ایم.
همان طور که در State Schema مشخص است ، بنده در این توصیف روح ها را یک مجموعه از مختصات تعریف کردم ، به این معنا که هر کدامشان در یک خانه از جدول هستند. حال ۴ operation Schema برای تعریف حرکت روح ها نوشتم یکی برای بالا رفتنشان ، یکی برای پایین رفت ، یکی برای راست رفتن و یکی برای چپ رفتن . در ادامه یکی از آن ها را توضیح میدهم.
یک مجموعه temp تعریف میکنیم از مختصات که معرف مجموعه جدیدی از مختصات روح هاست . حالا یا روح ها بالا میروند به طور همزمان و یا سر جایشان می ایستند . پس میگوییم به ازای هر مختصات از روح ها که داریم یک مختصات در مجموعه temp داریم که یا معادل خانه بالایی آن روح است و بدین معنی است که آن روح بالا می رود و یا معادل با همان خانه ای است که روح در آن قرار دارد به این معنا که روح سرجایش میماند و تکون نمی خورد.

GMoveUp

$\Delta GameTable$
$temp : \mathbb{P}\ Cell$

$\# temp = \# ghosts$
$\forall c : Cell \bullet c \in ghosts \Rightarrow \exists t : Cell \mid t \in temp \bullet t = c \lor$
$\quad (\exists i_1, i_2, j_1, j_2 : \mathbb{N} \mid (i_1, j_1) = c \land (i_2, j_2) = t \bullet i_2 = i_1 - 1 \land j_1 = j_2)$
$table(temp) = \{ path \}$
$temp \in dom\ table$
$table' = table$
$pacman' = pacman$
$ghosts' = temp$
$seeds' = seeds$
$length' = length$
$width' = width$

GMoveDown

$\Delta GameTable$

$temp : \mathbb{P}\ Cell$

$\# temp = \# ghosts$

$\forall c : Cell \bullet c \in ghosts \Rightarrow \exists t : Cell \mid t \in temp \bullet t = c \lor$
$\quad (\exists i_1, i_2, j_1, j_2 : \mathbb{N} \mid (i_1, j_1) = c \land (i_2, j_2) = t \bullet i_2 = i_1 + 1 \land j_1 = j_2)$

$table(temp) = \{ path \}$

$temp \in dom\ table$

$table' = table$

$pacman' = pacman$

$ghosts' = temp$

$seeds' = seeds$

$length' = length$

$width' = width$

GMoveRight

$\Delta GameTable$

$temp : \mathbb{P}\ Cell$

$\# temp = \# ghosts$

$\forall c : Cell \bullet c \in ghosts \Rightarrow \exists t : Cell \mid t \in temp \bullet t = c \lor$
$\quad (\exists i_1, i_2, j_1, j_2 : \mathbb{N} \mid (i_1, j_1) = c \land (i_2, j_2) = t \bullet i_2 = i_1 \land j_1 + 1 = j_2)$

$table(temp) = \{ path \}$

$temp \in dom\ table$

$table' = table$

$pacman' = pacman$

$ghosts' = temp$

$seeds' = seeds$

$length' = length$

$width' = width$

GMoveLeft

$\Delta GameTable$
$temp : \mathbb{P}\ Cell$

$\# temp = \# ghosts$
$\forall c : Cell \bullet c \in ghosts \Rightarrow \exists t : Cell \mid t \in temp \bullet t = c \lor$
$(\exists i_1, i_2, j_1, j_2 : \mathbb{N} \mid (i_1, j_1) = c \land (i_2, j_2) = t \bullet i_2 = i_1 \land j_1 - 1 = j_2)$
$table(temp) = \{ path \}$
$temp \in dom\ table$
$table' = table$
$pacman' = pacman$
$ghosts' = temp$
$seeds' = seeds$
$length' = length$
$width' = width$

حال در این بخش سعی کردم به نحوی حرکت تصادفی روح ها را بیان کنم

$MoveGhosts == \exists m : Move \bullet$
$(m = UP \land GMoveUp) \lor (m = DOWN \land GMoveDown) \lor$
$(m = RIGHT \land GMoveLeft) \lor (m = LEFT \land GMoveRight)$

در این بخش هم حالت بردن و باختن بازی را توصیف کرده ایم

Win

$\Xi GameTable$
$output! : Message$

$seeds = \varnothing$
$output! = win$

Defeated

$\Xi GameTable$
$output! : Message$

$pacman \in ghosts$
$output! = defeated$

$End == Win \lor Defeated$

$PacmanGame == MovePacman \lor EatSeed \lor MoveGhosts \lor End$