

<b>Formális módszerek (VIMIM100)</b>	<b>2009/2010. tanév II. félév</b>				<b>2010. március 22.</b>
<b>Első zárthelyi dolgozat, A csoport</b>	1.	2.	3.	4.	$\Sigma$
Név: _____					
NEPTUN kód: _____	12 pont	14 pont	16 pont	8 pont	50 pont

### 1. Kiskérdések (12 pont)

- 1.1. Hasonlítsa össze az implicit A operátorral kiegészített PLTL, a CTL és a CTL\* temporális logikák kifejezőképességét! 3 pont
- 1.2. Rajzolja fel és magyarázza el a tabló módszer alkalmazása során az U operátor felbontásának szabályát, és írja le, mikor juthatunk ezáltal *ellentmondásos* ághoz! 3 pont
- 1.3. Sorolja fel az ROBDD-k *legalább* 3 olyan tulajdonságát, amelyek *megkülönböztetik* a döntési fáktól (azaz ROBDD-re mindig teljesülnek, döntési fákra viszont nem)! 3 pont
- 1.4. Mit jelent és hogyan valósul meg UML állapotterképek esetén az „*események teljes feldolgoása* (run-to-completion)” elv? 3 pont

## 2. Bináris döntési diagramok (14 pont)

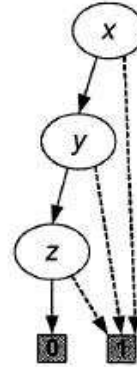
Adottak az alábbi logikai függvények:

$$f := \neg(\neg y \vee z) \vee (x \wedge y)$$

$$g := \text{lásd a jobboldali ábrát}$$

$$m := f \wedge g$$

Adott a  $g$  logikai függvény ROBDD-je:



(A szaggatott vonal tartozik a változó 0 behelyettesítéséhez, a folytonos pedig az 1-hez.)

- 2.1. Írja fel a  $g$  logikai függvény képletét algebrai alakban az ROBDD ábrából leolvasva! 3 pont
  
- 2.2. Rajzolja fel a lenti üres helyre az  $f$  logikai függvényt redukált rendezett bináris döntési diagram (ROBDD) alakban! 3 pont
  
- 2.3. Számítsa ki és ábrázolja az  $m$  logikai függvényt ROBDD alakban! A számítást közvetlenül az ROBDD-ken értelmezett műveletekkel végezze el! A függvényeket a  $g$  függvény ábráján megadott változósorrendben ( $x \rightarrow y \rightarrow z$ ) értékelje ki. 8 pont

### 3. Temporális logika és modellellenőrzés (16 pont)

Gyalogosok érkeznek egy olyan útkereszteződéshez, melynek forgalmát jelzőlámpa irányítja. A jelzőlámpa működését a baloldali Kripke struktúra (kapcsos zárójelben a logikai állítások, mint címkek találhatóak: *piros zöld, ki*); a gyalogosok magatartását a jobboldali tranzíciós rendszer (program) írja le.



3.1. Rajzolja fel először a gyalogos alrendszerének, majd az egész rendszernek (tehát a gyalogos és a jelzőlámpa együttesének) a Kripke struktúráját!

4 pont

3.2. Formalizálja LTL kifejezések segítségével az alábbi követelményeket!  
Vigyázat: úgy tekintse, hogy a követelmények formalizálásakor nem ismerjük magát a rendszert, csak a logikai állítások, azaz címkek halmazát! (válaszonként 2 pont)

6 pont

3.2-1. Ha a gyalogos érkezésekor piros a lámpa, akkor nem haladhat át mindaddig, amíg a lámpa zöldre nem vált.

3.2-2. A gyalogosnak nem kell örökké várakoznia az áthaladásra.

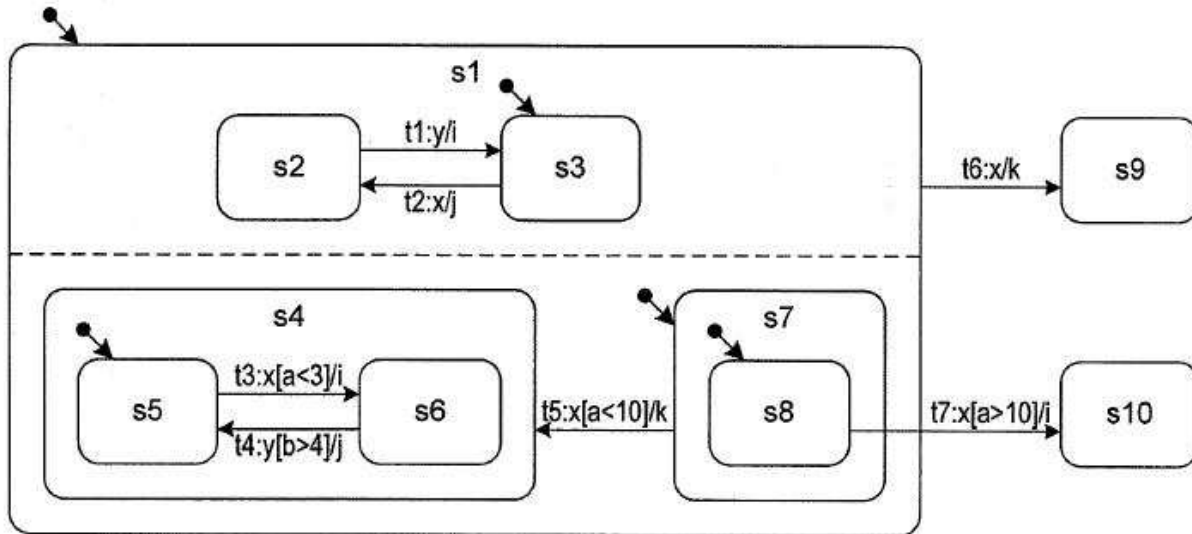
3.2-3. Ha egy gyalogos érkezésekor piros a lámpa, akkor a gyalogos a következő időpillanatban megáll.

3.3. Igaz-e a jelzőlámpa viselkedését leíró tranzíciós rendszerben (azaz a jelző Kripke struktúráját vizsgálva), hogy  $G(\neg \text{zöld} \vee X(\text{piros}))$ ? Válaszát a tabló módszer segítségével (külön lapon) elvégzett modellellenőrzés alapján adja meg!

6 pont

#### 4. UML állapottérképek (8 pont)

Tekintsük az alábbi állapottérképet, melyben minden  $s_k$  állapothoz tartoznak  $s_k.entry$  és  $s_k.exit$  akciók is, melyek az adott állapot belépési és a kilépési akcióit jelentik! A nyilakra írt kifejezések átmenetnév: trigger [őrfeltétel] / akció alakúak.



A rendszer a kezdő konfigurációjában van, az  $a$  változó értéke 8. Az esemény-sorrendezőtől egy „x” esemény érkezik.

- 4.1. Mi az állapottérkép kezdő állapotkonfigurációja? 1 pont
  
- 4.2. Melyek az engedélyezett állapotátmenetek? 1 pont
  
- 4.3. Mely engedélyezett állapotátmenetek állnak egymással konfliktusban? 1 pont
  
- 4.4. Hogy néz ki a tüzelhető állapotátmenetek halmaza (konfliktusfeloldás után)? 1 pont
  
- 4.5. Hogy néz(nek) ki a következő stabil állapotkonfiguráció(k)? (Ha több is lehetséges, mindet adja meg!) 2 pont
  
- 4.6. Milyen akciók hajtódnak végre tüzelés közben és milyen sorrendben? (Ha több tüzelés lehetséges, válasszon egy olyat, ami megváltoztatja az állapotkonfigurációt, arra adja meg!) 2 pont