Mintafeladatok az 1. zárthelyi dolgozathoz

1. Követelmények formalizálása temporális logika alkalmazásával

Egy vasúti kereszteződést biztosító fénysorompó viselkedését az állapotaihoz rendelt következő atomi kijelentésekkel jellemezzük: *{kikapcsolt, fehér, piros}*

A kereszteződéshez érkező autós viselkedését az állapotaihoz rendelt következő atomi kijelentésekkel jellemezzük: *{érkezik, körülnéz, megáll, áthalad}*

Formalizálja LTL kifejezések segítségével az alábbi követelményeket, amelyek az autós viselkedésére minden esetben vonatkoznak:

- Kikapcsolt állapotú fénysorompó esetén az autós körülnéz és a következő időpillanatban vagy áthalad, vagy megáll.
- Az autós előbb-utóbb át fog haladni a vasúti kereszteződésen.
- Ha egy autós érkezésekor piros a lámpa, akkor az autós addig nem halad át, amíg fehérre nem vált a fénysorompó.

2. Követelmények formalizálása temporális logika alkalmazásával

Egy bonyolult szimulációt futtató szerver állapotait a következő atomi kijelentésekkel jellemezzük: *{kikapcsolt, várakozó, bemelegítés, szimuláció}*

A szerverszoba hűtőberendezésének működését az állapotaihoz rendelt következő atomi kijelentésekkel jellemezzük: *{készenlét, normál, maximális}*

Formalizálja LTL kifejezések segítségével az alábbi követelményeket, amelyek a rendszer működésére minden esetben vonatkoznak:

- Ha egy adott pillanatban a szimuláció a hűtőberendezés készenléti állapota mellett zajlik, akkor a következő pillanatban a szerver várakozó állapotra kapcsol.
- Előbb-utóbb elkezdhető a szimuláció.
- Csak úgy hajtható végre szimuláció, ha volt bemelegítés a hűtőberendezés normál működése mellett.

3. Modellezés és modellellenőrzés

Egy informatikus hallgató „állapotait” aszerint különböztetjük meg, hogy éppen kávézik vagy nem, valamint éppen alszik vagy nem. A hallgató három tevékenységét különböztetjük meg: tanulás közben kávézik és nem alszik; ezután vizsgázik, ahol nem kávézik és nem is alszik; a vizsgázás után pihen, ekkor alszik és nem kávézik. A hallgató alapállapota a tanulás, amit a vizsgázásig nem is hagy abba. Tanulás nélkül a hallgató nem vizsgázik; a vizsgázás után közvetlenül pedig nem tanul (csak pihenés után).

- Rajzolja fel a hallgató itt leírt viselkedését modellező Kripke-struktúrát a hallgató kávézását és alvását figyelembe véve! Az egyes állapotokat jellemezze a következő atomi kijelentésekkel: *{pihen, tanul, vizsgázik}*

- Ellenőrizze a modellen, hogy a hallgató alapállapotából (ami a tanulás) kiindulva teljesül-e a következő CTL kifejezés: *E(¬vizsgázik U pihen)*! Válaszát indokolja meg!

4. Modellezés és modellellenőrzés

Egy informatikai rendszer két erőforrásból áll, ezek egy adatbázisszerver és egy alkalmazásszerver, amelyek kikapcsolt vagy bekapcsolt állapotban lehetnek. Az erőforrásokat hibamentes esetben egyszerre kapcsolják ki és be. Alaphelyzetben mindkét erőforrás ki van kapcsolva. A megfelelő üzemállapot az, amikor mindkét erőforrás be van kapcsolva. Ha az üzemállapotban az adatbázisszervert hiba következtében kikapcsolják, az rendszer szinten üzemképtelen állapotnak tekinthető. Ezután az alkalmazásszervert is kikapcsolják, majd mindkét erőforrás bekapcsolásával indítják újra a rendszert.

- Rajzolja fel a rendszer itt leírt működését modellező Kripke-struktúrát az egyes erőforrások bekapcsolását és kikapcsolását figyelembe véve! Az egyes állapotokat jellemezze a következő atomi kijelentésekkel: *{alaphelyzet, üzemállapot, üzemképtelen}*
- Ellenőrizze a modellen, hogy az üzemállapotot kezdőállapotnak tekintve teljesül-e a következő CTL kifejezés: *E(¬üzemképtelen U alaphelyzet)*! Válaszát indokolja meg!

5. Temporális logikai követelmények értelmezése

Indokolja meg, hogy következő ekvivalencia helyes-e: *F Stop V F Start ≡ F (Stop V Start)*, ahol *V* a logikai VAGY operátort jelöli, a *Stop* és *Start* pedig atomi kijelentések.

6. Temporális logikai követelmények értelmezése

Indokolja meg, hogy következő ekvivalencia helyes-e: *AG Stop ≡ not EF (not Stop)*, ahol *not* a logikai negálás operátort jelöli, a *Stop* pedig egy atomi kijelentés.