**Question**

Petri hálók tulajdonságai:

a) Egy *perzisztens* Petri háló két *tetszőleges* tranzíciója közül egyik sem tilthatja le a másikat

b) Egy Petri háló *fedhetősége* garantálja, hogy (ha *M* jelöli a *t* tranzíció tüzeléséhez szükséges minimális tokeneloszlást) ha *t* halott, akkor *M* lefedhető

c) Ha egy Petri háló *megfordítható*, akkor tetszőleges *M R(N, M0)* állapot visszatérő állapot

d) Ha egy Petri háló *biztos*, akkor egyúttal *korlátos* is

**Answer Key**

c, d

**Explanation**

Petri hálók tulajdonságai:

a) Egy *perzisztens* Petri háló két *tetszőleges* tranzíciója közül egyik sem tilthatja le a másikat

*A kiemelt tetszőleges szó magában foglalja az az esetet is, amikor nem engedélyezett egyik vagy másik tranzíció, márpedig ekkor nincs rájuk megkötés, letilthatják egymást.*

b) Egy Petri háló *fedhetősége* garantálja, hogy (ha *M* jelöli a *t* tranzíció tüzeléséhez szükséges minimális tokeneloszlást) ha *t* halott, akkor *M* lefedhető

*Hát,* t *akkor és csak akkor halott, ha* M***nem*** *fedhető le...*

c) Ha egy Petri háló *megfordítható*, akkor tetszőleges *M R(N, M0)* állapot visszatérő állapot

*Ha minden elérhető állapot visszatérő, akkor minden állapotból magába M0-ba is vissza tudunk térni*

d) Ha egy Petri háló *biztos*, akkor egyúttal *korlátos* is

*A biztos Petri háló a korlátos háló speciális esete, egységnyi korláttal*

2.

**Question**

1) Hogyan jellemezhető az *élő* tulajdonság?

a) Ha egy tranzíció *L2*-élő, akkor biztosan *L1*-élő is.

b) Ha egy *(N, M0)* Petri háló minden tranzíciója végtelen sokszor előfordul valamely tüzelési szekvenciában, akkor biztosan *L3*-élő, de nem biztosan *L4*-élő.

c) Ha egy *(N, M0)* Petri háló minden tranzíciója végtelen sokszor előfordul valamely tüzelési szekvenciában, akkor biztosan nincs benne deadlock.

d) Ha egy Petri háló elérhetőségi gráfja nem véges, akkor a háló biztosan *L3*-élő.

**Answer Key**

a, b

**Explanation**

1) Hogyan jellemezhető az *élő* tulajdonság?

a) Ha egy tranzíció *L2*-élő, akkor biztosan *L1*-élő is.

*Igen, az Li* *élőség mindig maga után vonja az Li-k élőséget*

b) Ha egy *(N, M0)* Petri háló minden tranzíciója végtelen sokszor előfordul valamely tüzelési szekvenciában, akkor biztosan *L3*-élő, de nem biztosan *L4*-élő.

*Az L3 élőség még mindig kezdőállapot függő, így az L4 élőség nem garantálható. Az állítás első része viszont pont L3 definíciója.*

c) Ha egy *(N, M0)* Petri háló minden tranzíciója végtelen sokszor előfordul valamely tüzelési szekvenciában, akkor biztosan nincs benne deadlock.

*A deadlockmentesség csak az L4 élő Petri hálók esetén garantált, az állítás pedig nem garantálja ezt (lásd előző pont)*

d) Ha egy Petri háló elérhetőségi gráfja nem véges, akkor a háló biztosan *L3*-élő.

*Attól, hogy az elérhetőségi gráf nem véges, még lehet, hogy egy tranzíciót egyáltalán nem sütünk el*

3.

**Question**

Mi igaz a *fedési gráfra*?

a) Végtelen állapottal rendelkező rendszernek vannak olyan állapotai, amelyekhez tartozó token eloszlások nem jelennek meg a rendszerhez tartozó fedési gráfban.

b) Ha egy fedési gráf egy csomópontjához tartozó token eloszlás vektorban a *pi* helyhez tartozó komponens *mi =* értékű, akkor mindig található ebből az állapotból induló olyan tüzelési szekvencia, amely a *pi* helyen levő tokenek számát eggyel növeli.

c) Ha egy Petri háló elérhetőségi gráfja nem korlátos, akkor az elérhetőségi és élőségi problémákat a fedési gráf alapján biztosan el lehet dönteni.

d) Végtelen állapottal rendelkező rendszert is modellezhet véges fedési gráffal rendelkező Petri háló.

**Answer Key**

d

**Explanation**

a) Végtelen állapottal rendelkező rendszernek vannak olyan állapotai, amelyekhez tartozó tokeneloszlások nem jelennek meg a rendszerhez tartozó fedési gráfban.

*A fedési gráf véges méretben kódolja a rendszer minden állapotát*

b) Ha egy fedési gráf egy csomópontjához tartozó token eloszlás vektorban a *pi* helyhez tartozó komponens *mi =* értékű, akkor mindig található ebből az állapotból induló olyan tüzelési szekvencia, amely a *pi* helyen levő tokenek számát eggyel növeli.

*Nem biztos, hogy pontosan eggyel növeli, lehet, hogy többel (pl. kettővel)*

c) Ha egy Petri háló elérhetőségi gráfja nem korlátos, akkor az elérhetőségi és élőségi problémákat a fedési gráf alapján biztosan el lehet dönteni.

*Nem korlátos esetben éppen, hogy nem lehet eldönteni (megoldani) e problémákat*

d) Végtelen állapottal rendelkező rendszert is modellezhet véges fedési gráffal rendelkező Petri háló.

*Sőt, mindig véges fedési gráffal modellezi.*

4.

**Question**

1) Egy véges kapacitású Petri háló ún. *kiegészítő helytranszformációjakor*

a) Minden helyhez még egy adminisztrációs helyet és egy korlátozó tranzíciót veszünk fel, tiltó élekkel összekötve

b) Minden tranzícióhoz rendelünk egy adminisztrációs helyet

c) A létrejövő társháló és az eredeti háló gyenge tüzelési szabályt feltételezve azonos tüzelési szekvenciát produkál

d) egy adminisztrációs hely kezdőállapota a (hozzátartozó) korlátozott kapacitású hely kihasználatlan kapacitása

**Answer Key**

d

**Explanation**

Egy véges kapacitású Petri háló ún. *kiegészítő helytranszformációjakor*

a) Minden helyhez még egy adminisztrációs helyet és egy korlátozó tranzíciót veszünk fel, tiltó élekkel összekötve

*Minden helyhez csak egy adminisztrációs helyet veszünk fel. Korlátozó tranzíció, hmmm :)*

b) Minden tranzícióhoz rendelünk egy adminisztrációs helyet

*Minden helyhez....*

c) A létrejövő társháló és az eredeti háló gyenge tüzelési szabályt feltételezve azonos tüzelési szekvenciát produkál

*Az eredeti háló erős és a társháló gyenge tüzelési szabályát feltételezve lesznek azonosak a tüzelési szekvenciák*

d) Egy adminisztrációs hely kezdőállapota a (hozzátartozó) korlátozott kapacitású hely kihasználatlan kapacitása

*Igen, így inicializáljuk a társhálót.*

5.

**Question**

1) Melyek az ábrán látható Petri hálójának minimális P invariánsai (olyan P invariánsok, melyekből minden más P invariáns előállítható, de maguk már nem állnak elő P invariánsok összegeként)?

[0.gif]

a) p3+p5; p2+p6; p1+p4

b) p2+p3+p5+p6; p1+p3+p4+p5

c) p2+p3+p5+p6; p1+p4

d) p2+p6; p1+p4

**Answer Key**

a

**Explanation**

*a) a = 1, b = 2, c = -1, d = 0*

*b) a = -1, b = 1, c = 1, d = 0*

*c) a = 1, b = -1, c =0, d = 1*

d) a = 0, b = 0, c = 0, d = 1

*A példában minden sorra és minden oszlopra igaznak kell lennie, hogy az elemek összege 0. Ez alapján először c és d, majd a és b is meghatározható. Végezetül, keressük meg az ábrán az egyes helyeket és tranzíciókat.*

6.

**Question**

Jelöld meg az alábbi kérdéshez tartozó állítások közül azokat, amelyek helyes válasz adnak az adott kérdésre! Figyelem: több is helyes lehet a megadott állítások közül! Segítség: minden kérdésnél van legalább egy igaz állítás.

Mi igaz a tranzíciók prioritására?

**Explanation**

Egy idõzítetlen átmenet magasabb prioritású mint a vele egyidõben engedélyezett idõzített átmenet.

Egy prioritási szinten belül az aktivizálandó tüzelés kiválasztása kötött sorrendben történik.

Egy adott token eloszlás esetén az engedélyezett átmenetek között levõ a feleakkora prioritású átmenetek feleakkora valószínûséggel tüzelhetnek mint a náluk kétszer akkora prioritással rendelkezõ engedélyezett átmenetek.

Ha egy helybõl egy kisebb és egy nagyobb prioritású idõzítetlen tranzícióba egyaránt vezet él akkor nincs olyan tüzelési szekvencia amelyben a kisebb prioritású tranzíció tüzelése megelõzi a nagyobb prioritású tranzíció tüzelését.

**Answer Key**

Egy idõzítetlen átmenet magasabb prioritású, mint a vele egyidõben engedélyezett idõzített átmenet.

**Explanation**

Egy időzítetlen átmenet magasabb prioritású, mint a vele egyidőben engedélyezett időzített átmenet.

*Így definiáltuk, hogy ne adódjon újabb konfliktushelyzet*

Egy prioritási szinten belül az aktivizálandó tüzelés kiválasztása kötött sorrendben történik.

*Nem, mert nemdeterminisztikusan*

Egy adott token eloszlás esetén az engedélyezett átmenetek között levő feleakkora prioritású átmenetek feleakkora valószínűséggel tüzelhetnek, mint a náluk kétszer akkora prioritással rendelkező engedélyezett átmenetek.

*Előbb a magasabb prioritásúak tüzelnek, csak utána az alacsonyabbak*

Ha egy helyből egy kisebb és egy nagyobb prioritású időzítetlen tranzícióba egyaránt vezet él, akkor nincs olyan tüzelési szekvencia, amelyben a kisebb prioritású tranzíció tüzelése megelőzi a nagyobb prioritású tranzíció tüzelését.

*Nem biztos, hogy egyszerre lesznek engedélyezettek, így a kisebb prioritású lehet akkor is engedélyezett, amikor a nagyobb prioritású nem az.*

7.

**Question**

A *részleges döntési technika*

a) Minden esetben megad egy ellenpéldát, ha az létezik

b) Képes megmutatni, ha a Petri háló állapotegyenlete nem megoldható

c) Létrehoz egy szupermodellt, amely egy megkívánt tulajdonság szükséges feltételének kielégíthetetlenségét próbálja garantálni

d) Egy olyan szupermodellt állít elő, amely nem hoz be az eredeti rendszer szempontjából hamis megoldásokat

**Answer Key**

b

**Explanation**

A *részleges döntési technika*

a) Minden esetben megad egy ellenpéldát, ha az létezik

*Ha nem létezik ellenpélda, azt képes bizonyítani*

b) Képes megmutatni, ha a Petri háló állapotegyenlete nem megoldható

*Ez lenne a célja.*

c) Létrehoz egy szupermodellt, amely egy megkívánt tulajdonság szükséges feltételének kielégíthetetlenségét próbálja garantálni

*A szükséges feltétel* ***tagadásának*** *kielégíthetetlenségét próbálja garantálni a szupermodell*

d) Egy olyan szupermodellt állít elő, amely nem hoz be az eredeti rendszer szempontjából hamis megoldásokat

*Sajnos a szupermodell behozhat hamis megoldásokat*

8.

**Question**

Egy alkalmazás kommunikációs csatornája az alkalmazás és a hálózat közötti adatforgalmat bonyolítja le.

• Az alkalmazás által küldött adatcsomagot a csatorna egy *m* egyedi azonosítóval és egy *dat* típusjelzéssel látja el, majd az elküldendő üzenetek közé teszi. Mindezen túl, az üzenet nyugtájának megérkezéséig el is tárolja azt, felkészülve az esetleges üzenetvesztésre.

• Ha a csatorna *dat* típusú üzenetet kap, akkor azt továbbítja az alkalmazás felé, és egy, a kapott üzenet *m* sorszámával ellátott, *ack* típusú üzenetet helyez el a továbbítandó üzenetek közé.

• Amennyiben pedig egy általa korábban küldött adatcsomag nyugtája (*ack* formájában) érkezik meg, akkor az adatcsomagot kiveszi az időszakosan tárolt üzenetek közül.

• Amennyiben azonban egy *to* (timeout) típusú *m* azonosítójú üzenet érkezik egy (szintén a hálózatra kapcsolódó) időzítőtől, akkor a csatorna kikeresi a tárolt üzenetek közül a megfelelő sorszámút, és újraküldi.

Miért NEM felel meg a feladat specifikációjának az alábbi színezett Petri hálós modell? (Több válasz is lehetséges)

**Answer Key**

a, d

**Explanation**

*Az ábrából kihagytunk egy élet (a jobb alsó hely és a drop\_stored tranzíció között), amely csak nyugtázás esetén dobná el a tárolt üzenetet.*

a) A tárolt üzenetek nyugta nélkül is törlődnek

*Ez volt a szándékos ábrarontás célja*

b) Egy nyugta küldése megelőzheti a hozzátartozó üzenet megérkezését

*Nem, nyugtát csak akkor állít ki a jobb felső hely, ha a "dat" jelzéssel ellátott üzenetet már feldolgozta*

c) Üzenetek elvesztése esetén is küldünk nyugtát

*Nem küldünk, lásd előző pont*

d) A nyugtázott üzenetek nem törlődnek a rendszerből

*Ez sajnos igaz, nem biztos, hogy törlődnek a rendszerből, mivel a nyugtázás és törlés egymástól függetlenül működik.*

9.

**Question**

1) Adott az ábrán látható Petri háló, és WT szomszédossági mátrixa. Mely számokat kell a kitöltetlen helyekre írnunk?

*a) a = 1, b = 2, c = -1, d = 0*

*b) a = -1, b = 1, c = 1, d = 0*

*c) a = 1, b = -1, c =0, d = 1*

d) a = 0, b = 0, c = 0, d = 1

**Answer Key**

c

**Explanation**

Adott az ábrán látható Petri háló, és WT szomszédossági mátrixa. Mely számokat kell a kitöltetlen helyekre írnunk?

[2a.gif, 2b.gif]

*a) a = 1, b = 2, c = -1, d = 0*

*b) a = -1, b = 1, c = 1, d = 0*

*c) a = 1, b = -1, c =0, d = 1*

d) a = 0, b = 0, c = 0, d = 1

*A példában minden sorra és minden oszlopra igaznak kell lennie, hogy az elemek összege 0. Ez alapján először c és d, majd a és b is meghatározható. Végezetül, keressük meg az ábrán az egyes helyeket és tranzíciókat.*

10.

**Question**

Mely állítások igazak a *szomszédossági mátrixra*?

a) A szomszédossági mátrixnak annyi sora van, ahány hely van a Petri hálóban, és annyi oszlopa, ahány él van a helyek között.

b) A W^T szomszédossági mátrixszal rendelkező Petri hálóban a *t* tranzíció tüzelésének hatására az M0 kezdőállapotból a rendszer a Mi=M0+W^T\*[epsilon]^T állapotba megy át, ahol [epsilon]^T a *t* tranzíciónak megfelelő egységvektor.

c) A szomszédossági mátrix egy *w(t, p)* eleme a *t* tranzició tüzelésekor a hozzá kapcsolódó *p* helyen levő tokenek száma.

d) A Petri hálóban levő egységnyi súlyú hurokélek (amikor egy *t* tranzició a hozzá kapcsolódó *p* helynek egyszerre bemeneti és kimeneti tranzíciója) a szomszédossági mátrixban negatív előjelű elemekként jelennek meg.

**Explanation**

Error on page.