

1. Állapítsa meg, hogy az ASN.1 BER kódolás szabályai szerint az UzenetTipus egyértelműen dekódolható-e? Válaszát indokolja! 3p
Ha nemleges választ adott, tegyen javaslatot a hiba kijavítására!

Nem.

sorszám és típus komponensek címkeje azonos és az első opcionális

Javítás:

- modul fejlécben AUTOMATIC TAGS

- valamelyik komponensnek kézi címke

pl. sorszám [0] INTEGER OPTIONAL

2. Melyik állítás igaz az UzenetTipus típus komponenseire (mezőire)? 2p

- a) Mindhárom komponens az ASN.1 modul első verziójában definiáltuk és a típust a későbbiekben nem bővíthetjük.
- b) Mindhárom komponens az ASN.1 modul első verziójában definiáltuk. Az újabb verziókban hozzáadott komponenseket a típus és tartalom közé illeszthetjük be.
- c) sorszám és típus komponenseket az első, a tartalom komponensét a második verzióban definiáltuk. A jövőben új komponenseket tetszésünk szerint, a tartalom elé vagy mögé is adhatunk.
- d) A. sorszám és típus komponenseket az első, a tartalom komponensét a második verzióban definiáltuk. Új komponenseket csak a tartalom mögé, azaz a típus legvégére szúrhatunk be.

3. Melyik állítás igaz a t_alapUzenet template (üzenetminta) tartalom mezőjére? 2p

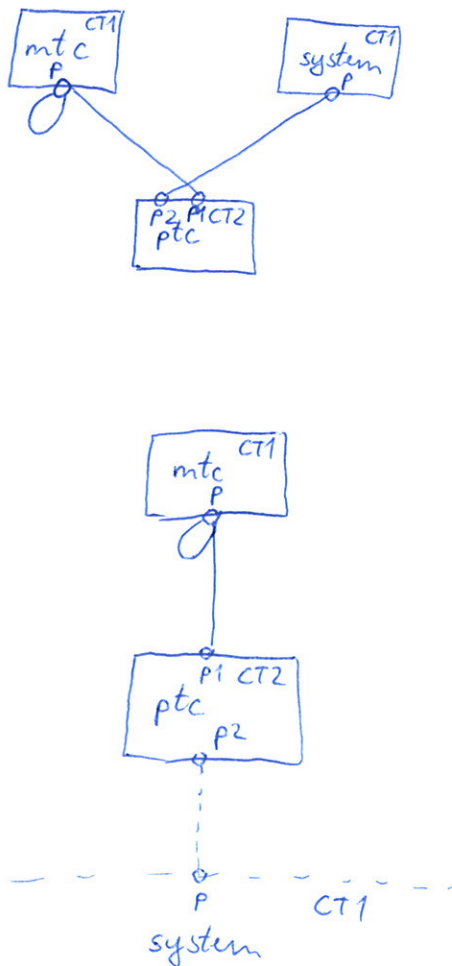
- a) Olyan, legalább 3 oktett hosszúságú füzérekre illeszkedik, amelyekben (bárhon) megtalálható a 12 és 56 minta.
- b) Olyan, legfeljebb 3 oktett hosszúságú füzérekre illeszkedik, amelyekben (bárhon) megtalálható a 12 és 56 minta.
- c) Olyan, pontosan 3 oktett hosszúságú füzérekre illeszkedik, amelyek első ill. utolsó oktettje rendre 12 és 56 értékű.
- d) Olyan, legalább 3 oktett hosszúságú füzérekre illeszkedik, amelyek első ill. utolsó oktettje rendre 12 és 56 értékű.
- e) Nem létezik a mintára illeszkedő oktett-füzér.

4. Készítsen `t_modositottUzenet`-tel megegyező tartalmú, nem származtatott üzenetmintát (template-et). Az új template-et nevezze `t_egyszeruUzenet`-nek!

2p

```
template UzenetTipus t_egyszeruUzenet :=  
{  
    sorszam := 1,  
    tipus := 0,  
    tartalom := '1200000056'0  
}
```

5. Rajzolja fel a tc tesztet futása során létrejövő teszt konfigurációt (teszt-komponenseket, portokat és a köztük lévő kapcsolatokat)! 5p



6. Mi történik, ha a fenti tc teszteset send operációjából kitöröljük a to kifejezést (tehát egyszerűen csak ezt írjuk: `p.send(t_modositottUzenet);`)? 2p

- a) Mivel a p portnak csak egyetlen kapcsolata van, ez nem okoz semmilyen problémát.
- b) A művelet végrehajtásakor futási idejű hiba (dynamic testcase error) történik, mert a p portnak egyidejűleg több kapcsolata is van.
- c) Az üzenet a port összes kapcsolatán elküldésre kerül (broadcasting).
- d) A futtatórendszer véletlenszerűen választ, hogy melyik kapcsolaton keresztül küldje ki az üzenetet.
- e) Mivel nem egyértelmű, hogy melyik kapcsolaton keresztül szeretnénk küldeni, üzenetküldés nem történik, de a végrehajtás folytatódik a következő utasítással.

7. Mennyi idő alatt és milyen ítélettel fejeződik be a tc teszteset futása? Feltételezzük, hogy a tesztelendő rendszer (SUT) minden beérkező üzenetet változtatás nélkül azonnal visszaküld a feladónak. 2p

- a) Nagyon gyorsan, tipikusan a másodperc törtrésze alatt lefut. Az eredmény sikeres (pass) lesz.
- b) Legalább 10 másodpercig tart és eldönthetetlen (inconc) ítélettel ér véget.
- c) A teszt-rendszer holtpontra kerül, emiatt soha nem fejeződik be.
- d) Indítás után nagyon hamar futási idejű hibával áll le. Emiatt az ítélet hiba (error) lesz.

Σ18p