

Operációs rendszerek II.

Holtpont



Holtpont (*deadlock*) fogalma

- A folyamatok egy csoportja olyan eseményre vár, amelyet egy másik, ugyancsak várakozó folyamat tud előidézni.
- Esemény:
 - tipikusan erőforrás felszabadulása.

Kiéheztetés és a holtpont különböző fogalmak!

Rendszer modell I.

- Véges számú erőforrás.
- Erőforrás osztályok:
 - csereszabatos erőforrások.
- Elvehető (preemptable) és nem elvehető (non-preemptable) erőforrások.
- A rendszer állapotának leírása: erőforrás-használati gráffal (*resource allocation graph*).

Dr. Benyó Balázs
Operációs rendszerek II.

Holtpont példa

P1 folyamat

...

Lefoglal(M)

<mágnesszalag használata>

Lefoglal(NY)

<nyomtató és mágnesszalag együttes
használata>

Felszabadít(M)

<nyomtató használata>

Felszabadít(NY)

...

Dr. Benyó Balázs
Operációs rendszerek II.

P2 folyamat

...

Lefoglal(NY)

<nyomtató használata>

Lefoglal(M)

<nyomtató és mágnesszalag együttes
használata>

Felszabadít(NY)

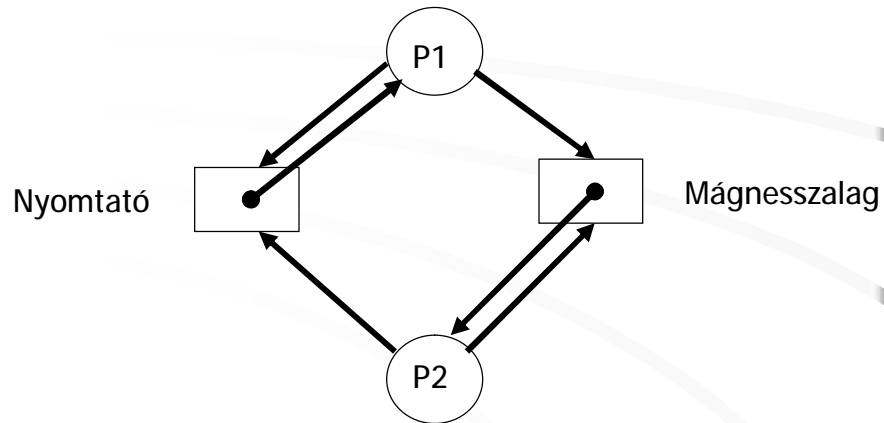
<mágnesszalag használata>

Felszabadít(M)

...

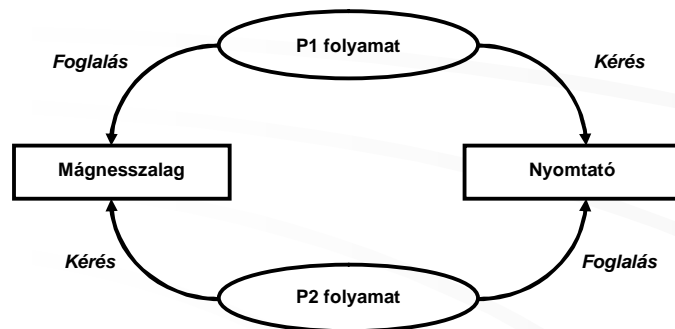
Dr. Benyó Balázs
Operációs rendszerek II.

Holtpont kialakulása



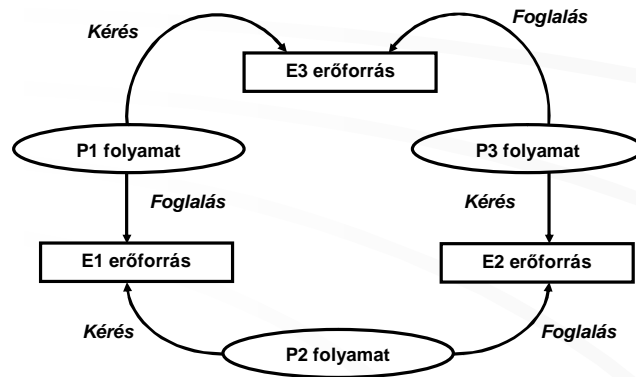
Dr. Benyó Balázs
Operációs rendszerek II.

Egyszerű holtpont szituáció



Dr. Benyó Balázs
Operációs rendszerek II.

Holtpont három folyamattal



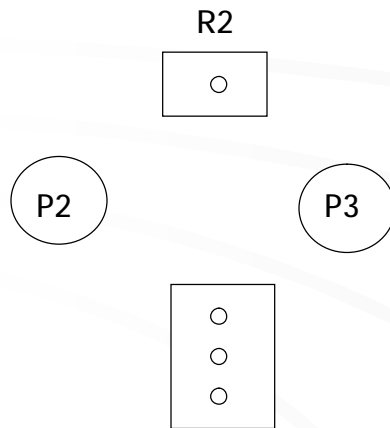
Dr. Benyó Balázs
Operációs rendszerek II.

Erőforrás-használati gráf

- Erőforrások
- Csereszabatos erőforrások csoportja
- Folyamatok
- Erőforrás igénylés
- Erőforrás foglalás

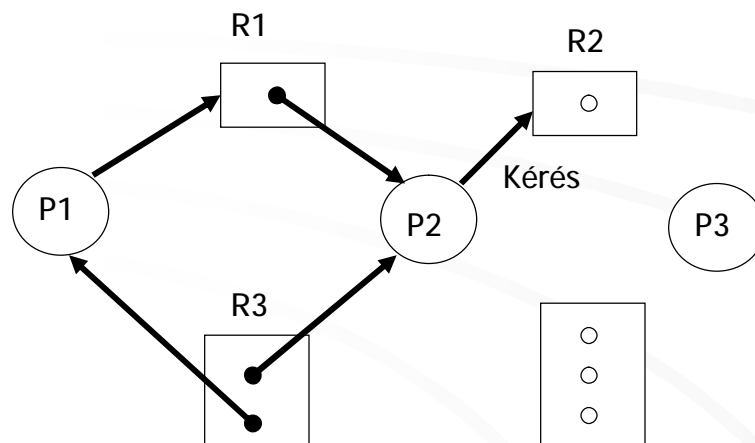
Dr. Benyó Balázs
Operációs rendszerek II.

Erőforrás-használati gráf



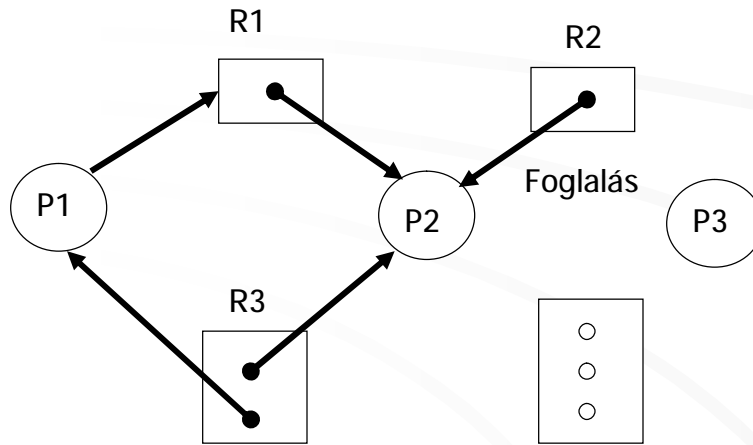
Dr. Benyó Balázs
Operációs rendszerek II.

Erőforrás-használati gráf



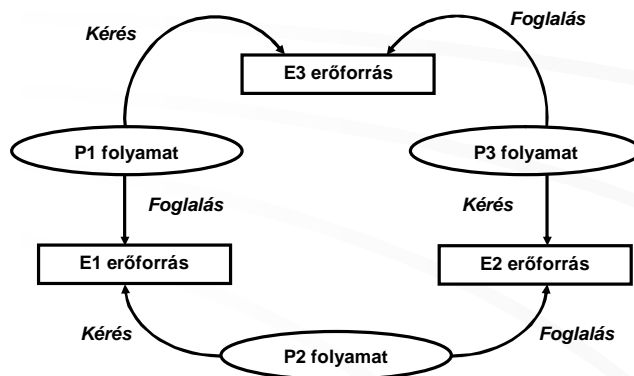
Dr. Benyó Balázs
Operációs rendszerek II.

Erőforrás-használati gráf



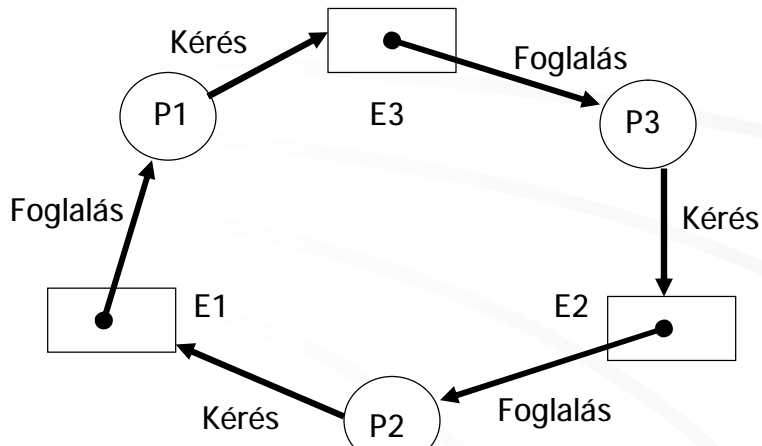
Dr. Benyó Balázs
Operációs rendszerek II.

Holtpont három folyamattal



Dr. Benyó Balázs
Operációs rendszerek II.

Erőforrás-használati gráf



Dr. Benyó Balázs
Operációs rendszerek II.

Rendszer modell II.

Erőforrás használat lépései:

- Igénylés:
 - Ha az igény nem teljesíthető, a folyamat várakozik.
- Az erőforrás kizárólagos használata.
- Az erőforrás felszabadítása:
 - Valamelyik várakozó továbbengedése.

Dr. Benyó Balázs
Operációs rendszerek II.

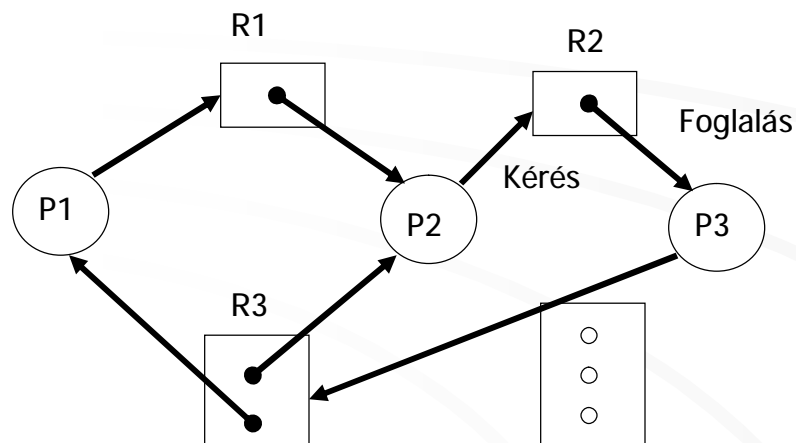
A holtpont kialakulásának feltételei I.

A holtpont kialakulásának *szükséges* feltételei:

- kölcsönös kizárás (erőforrás használat);
- foglalva várakozás;
- nem elvehető erőforrások;
- körkörös várakozás.

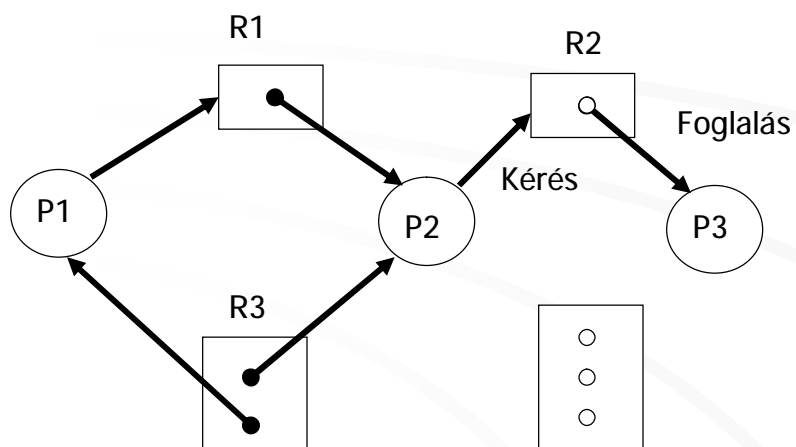
Dr. Benyó Balázs
Operációs rendszerek II.

Holtpont



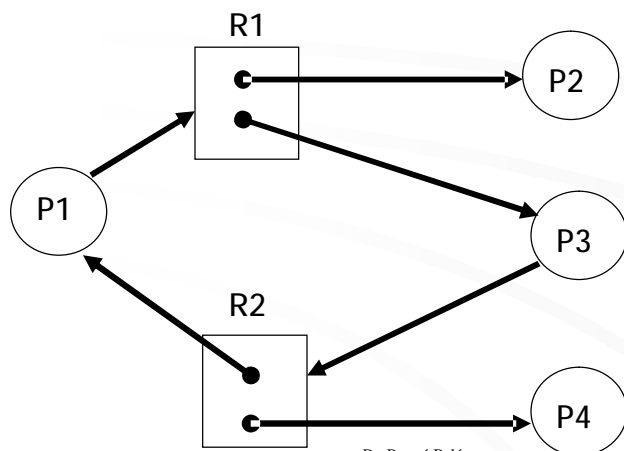
Dr. Benyó Balázs
Operációs rendszerek II.

Nincs holtpont



Dr. Benyó Balázs
Operációs rendszerek II.

Nincs holtpont



Dr. Benyó Balázs
Operációs rendszerek II.

A holtpont kialakulásának feltételei II.

A négy szükséges feltétel egyben *elégséges* feltétel is, ha minden erőforrás osztályban csak *egyetlen* erőforrás van.

Dr. Benyó Balázs
Operációs rendszerek II.

Holtpont kezelési stratégiák

- Erőforrás használati szabályokkal biztosítani, hogy holtpont ne alakuljon ki:
 - Holtpont megelőzés (deadlock prevention)
 - Holtpont elkerülés (deadlock avoidance)
- Csak a holtpont kialakulásánál avatkozunk be:
 - Holtpont felismerés (deadlock recognition)
 - Holtpont felszámolása (deadlock recovery)

Dr. Benyó Balázs
Operációs rendszerek II.

A holtpont megelőzése I.

- A holtpont kialakulásának valamelyik szükséges feltételének kizárása:
 - *Kölcsönös kizárást* nem lehet kiküszöbölni.
 - *Foglalva várakozás kizárása:*
Egy folyamat csak akkor kérhet új erőforrást, ha nem tart lefoglalva másikat. Alternatívák:
 - Futás elején lefoglalja az összes erőforrást.
 - Erőforrás-foglalás előtt a foglalt erőforrásokat felszabadítja.

Dr. Benyó Balázs
Operációs rendszerek II.

A holtpont megelőzése II.

- *Nem elvehető erőforrások* kezelése:
 - erőforrások elvétele egyes folyamatoktól.
- *Körkörös várakozás kizárása:*
 - erőforrások megszámozása,
 - erőforrások csak sorrendben igényelhetők.

Dr. Benyó Balázs
Operációs rendszerek II.

A holtpont elkerülése

- Az erőforrások óvatos allokálásával elkerülhető a holtpont kialakulása.
- Megvalósítás feltétele:
a folyamatok erőforrás-igényéről kiegészítő információval kell rendelkezni:
 - a folyamatok erőforrás-osztályonkénti maximális igénye.

Dr. Benyó Balázs
Operációs rendszerek II.

Rendszer modell

- Feltételezés:
A folyamat erőforrás igényének kielégítése után véges időn belül visszaadja az összes erőforrást.

Dr. Benyó Balázs
Operációs rendszerek II.

A holtpont elkerülésének módszere

- Az erőforrás igényt csak akkor teljesítjük, ha az így kialakult rendszer biztonságos állapotban marad.
- Biztonságos állapot:
Található olyan folyamat végrehajtási sorrend, hogy az aktuálisan futó folyamat maximális igénye kielégíthető.
- Megvalósítás:
Bankár algoritmus.

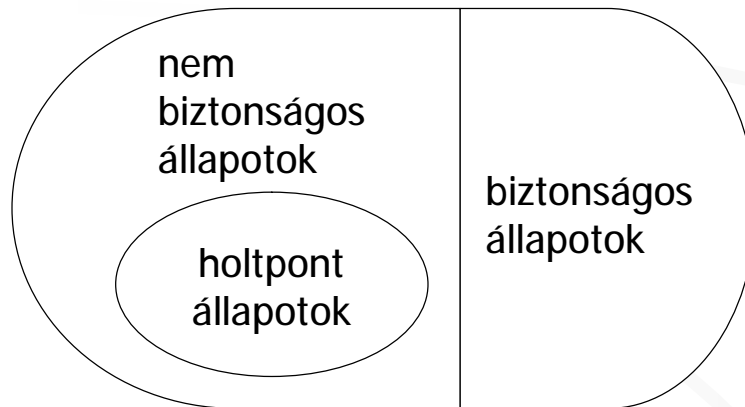
Dr. Benyó Balázs
Operációs rendszerek II.

Holtpont elkerülése

- Folyamatokról erőforrás használatáról többletinformációval kell rendelkezni:
 - Minden erőforrásosztályból egyidőben maximálisan szükséges erőforrás példányok szám.
- Bankár algoritmus:
 - Csak azokat a folyamatokat engedi futni, melyek erőforrásigénye kielégíthető ha az összes erőforrást igényli, amire szüksége van.

Dr. Benyó Balázs
Operációs rendszerek II.

Biztonságos és holtpont állapotok viszonya



Dr. Benyó Balázs
Operációs rendszerek II.

A holtpont felismerése I.

- Az erőforrás-használati gráf elemzése alapján.
- Gráf redukciós algoritmus:
 - Minden olyan folyamat kiválasztása, amelynek a jelenlegi igényei a szabad erőforrásokkal kielégíthetők.
 - A folyamat által lefoglalt erőforrásokat visszaadjuk (optimista algoritmus).
 - Újra keresünk kielégíthető folyamatot, ha nincs ilyen, de maradtak kielégítetlen folyamatok, akkor holtpont van.

Dr. Benyó Balázs
Operációs rendszerek II.

A holtpont felismerése II.

- Hátrány:
 - Az algoritmus viszonylag lassú.
- Mikor fusson?
 - Minden erőforrás igény teljesítésekor.
 - Meghatározott időközönként.

Dr. Benyó Balázs
Operációs rendszerek II.

A holtpont felszámolása I.

- Folyamatok megszüntetésével:
 - Minden holtpontban lévő folyamatot megszüntetünk.
 - Csak néhány folyamatot szüntetünk meg.
 - A megszüntetendő folyamat kiválasztásához paraméterek pl.:
 - Hány holtpont hurokban szerepel.
 - Mekkora a prioritása.
 - Mennyi ideje fut (várhatóan mennyi ideig futna még).
 - Milyen erőforrásokat tart lefoglalva.
 - Milyen erőforrásokat igényelne még.

Dr. Benyó Balázs
Operációs rendszerek II.

A holtpont felszámolása II.

- Erőforrások elvételével.
Probléma: melyik folyamattól és milyen erőforrásokat vegyük el.
- A futási eredmény megőrzéséhez szükséges ellenőrzési pontok (*checkpoint*) definiálása és a visszaállítás (*rollback*) használata.

Dr. Benyó Balázs
Operációs rendszerek II.

Kombinált holtpont-kezelő stratégiák

- Különböző erőforrás-típusokhoz más-más stratégiát használhatunk, pl.:
 - Belső rendszer-erőforrások: sorrendi foglalás.
 - Központi tár: erőforrás elvétele.
 - Kötegelt rendszerekben, előre megadott erőforrás szükséglet: holtpont elkerülés.
 - Tárcsere terület a háttértáron: előzetes foglalás.

Dr. Benyó Balázs
Operációs rendszerek II.

Megszakítások kezelése

- Megszakítások kezelése: megszakítás kiszolgálásainak lépései.
- Alapvető módszerek:
 - kiszolgáló rutin (nincs környezetváltás),
 - kiszolgáló folyamat (mindig környezetváltás)
- Köztes megoldás, kiszolgáló rutin, a várakozó folyamatok átütemezésének kezdeményezése.

Dr. Benyó Balázs
Operációs rendszerek II.

Bankár algoritmus - példa

Bankár algoritmus

	MAX				VAN				10	8	5	13						
	E1	E2	E3	E4	AKTUÁLIS	E1	E2	E3	E4					E1	E2	E3	E4	
F1	5	2	4	3		1	2	1	3					4	0	3	0	HAMIS
F2	7	5	2	7		2	2	2	3					5	3	0	4	HAMIS
F3	2	3	1	2		1	3	0	2					1	0	1	0	IGAZ
F4	3	6	5	2		3	1	1	1					0	5	4	1	HAMIS
					SUM	7	8	4	9									
					REST	3	0	1	4									
F1	5	2	4	3		1	2	1	3					4	0	3	0	HAMIS
F2	7	5	2	7		2	2	2	3					5	3	0	4	HAMIS
F3	2	3	1	2		0	0	0	0					2	3	1	2	IGAZ
F4	3	6	5	2		3	1	1	1					0	5	4	1	HAMIS
					SUM	6	5	4	7									
					REST	4	3	1	6					Nincs BIZTONSÁGOS állapotban!				