



**OSTRAVSKÁ UNIVERZITA**  
PŘÍRODOVĚDECKÁ FAKULTA

# Petri Nets

Ing. Pavel Smolka, Ph.D.

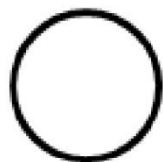
**Petriho síť je automat, který definuje chování systému posloupnostmi událostí a odpovídajícími stavovými změnami systému.**

**Registrace dosažení určitého stavu –značení místa**

- Nezáporná celočíselná informace.
- Grafické znázornění černými tečkami.
- Místo obsahuje značku –podmínka platí.
- Místo neobsahuje značku –podmínka neplatí.
- Značení může být omezené -omezená kapacita.



# Grafické znázornění



Place

(a)



Transition

(b)



Arc

(c)



Token

(d)



normal arc



inhibitory arc



test arc



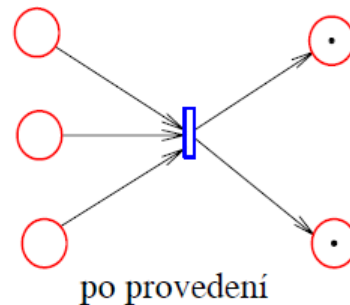
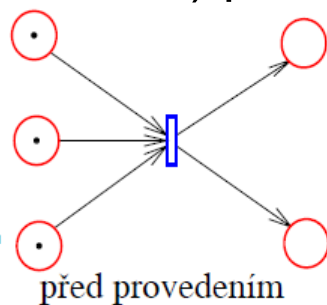
# Typy

- C/E (Condition / Event) Petriho sítě.
- P/T (Place / Transition) Petriho sítě.
- P/T Petriho sítě s inhibičními hranami.
- P/T Petriho sítě s prioritami.
- TPN Časované (Timed) Petriho sítě.
- CPN Barevné (Coloured) Petriho sítě.
- HPN Hierarchické (Hierarchical) Petriho sítě.



# C/E (Condition / Event) Petriho síť

- Podmínky (conditions) zobrazované kroužky,
- události (events) zobrazované obdélníky (případně úsečkami),
- šipky vedoucí od podmínek k událostem,
- šipky vedoucí od událostí k podmínkám,
- tokeny (zobrazenými tečkami v kroužcích podmínek) indikujícími logický stav (pravdivost) podmínek.



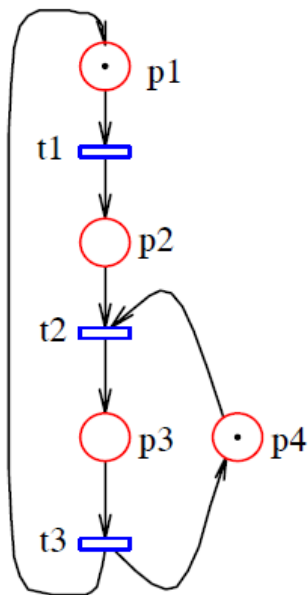
# Změny stavů C/E Petriho sítí

Ke změnám stavu sítě dochází uskutečňováním událostí:

- událost může nastat, jsou-li všechny její vstupní podmínky splněny a současně všechny její výstupní podmínky nesplněny - takovou událost nazýváme proveditelnou,
- provedena může být pouze proveditelná událost, proveditelná událost může také zůstat neprovedena,
- po provedení proveditelné události jsou všechny její výstupní podmínky splněny a všechny její vstupní podmínky nesplněny.



# Příklad č. 1 - cyklický proces, který čas od času potřebuje využívat nějaký zdroj.



**p1:** proces činný bez potřeby zdroje

**p2:** proces čeká na přidělení zdroje

**p3:** proces využívá zdroj

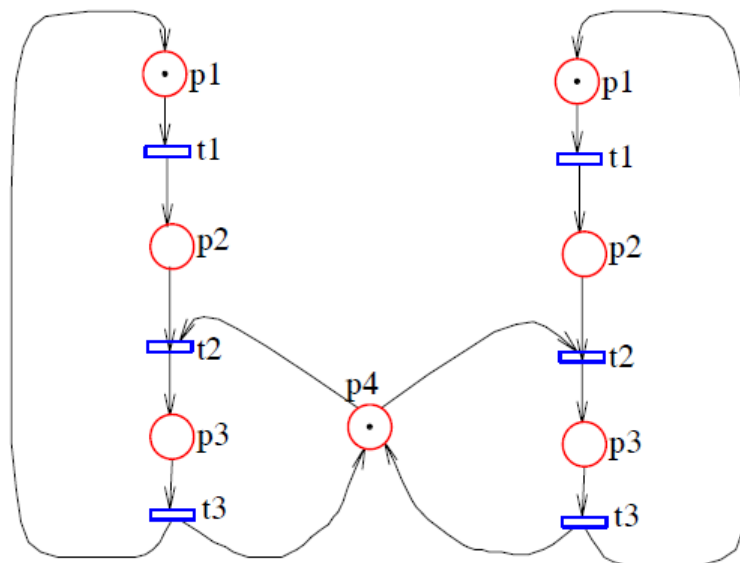
**p4:** zdroj není využíván

**t1:** vznik požadavku na zdroj

**t2:** počátek využívání zdroje

**t3:** konec využívání zdroje

## Příklad č. 2 - dva nekonečné cyklické procesy, které čas od využívají společný zdroj



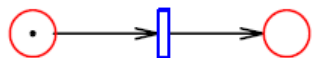


# P/T (Place / Transition) Petriho síť

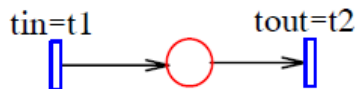
- P/T Petriho síť je trojice  $N = (P, T, F)$
- P a T jsou disjunktní množiny
- F je binární relace
- P je množina míst (Places) sítě N
  - obsahují stavovou informaci ve formě značek (tokenů)
- T je množina přechodů (Transitions) sítě N
  - vyjadřují možné změny stavů (vzory možných událostí)
- F je toková relace (Flow relation) sítě N
  - určují logické vazby



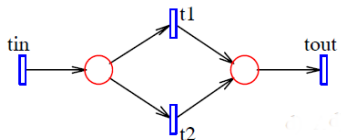
# Elementární programy PN



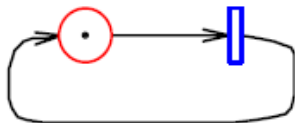
Jednorázový elementární program



Sekvence

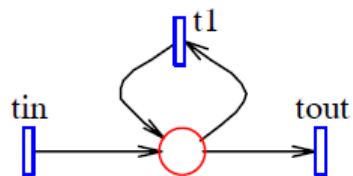


Alternativa

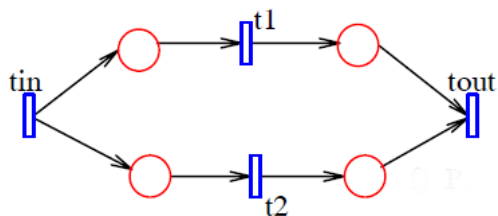


Nekončící elementární program

# Elementární programy PN



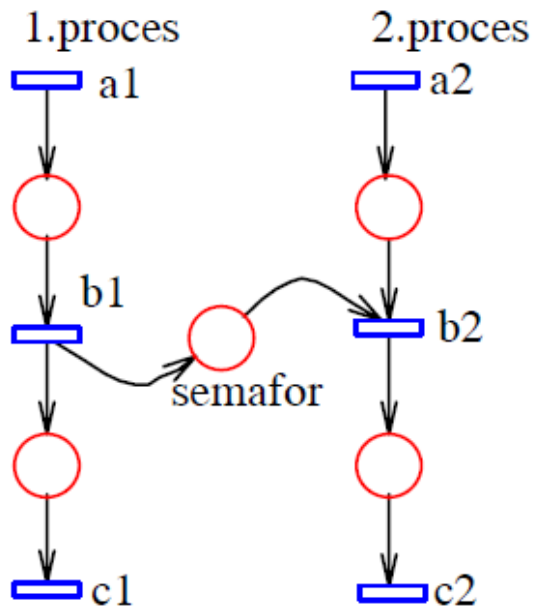
Cyklus (iterace)



Paralelismus

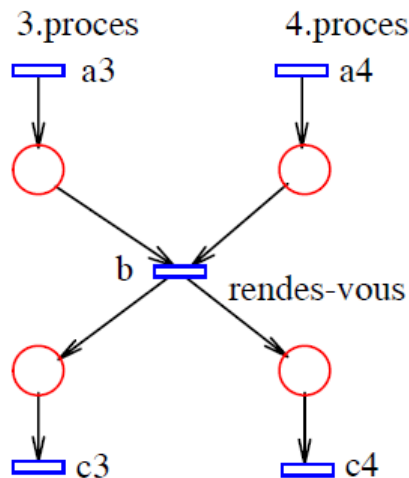


# Příklad synchronizace procesů pomocí semaforu



- 1. proces nastavuje semafor na "volno" pro 2. proces
- událost b2 nemůže nastat dříve než událost b1
- událost b2 nastavuje semafor na "stát"

# Příklad synchronizace procesů pomocí rendez-vous



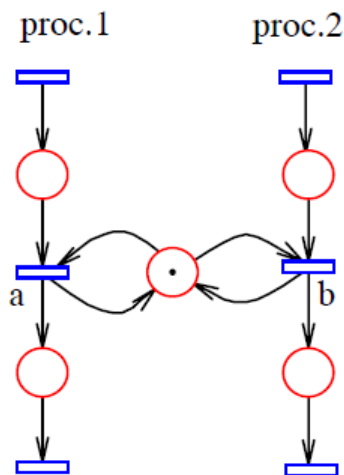
- Obrázek představuje schůzku (rendes-vous) procesů 3.a 4.
- kterákoliv z událostí c3,c4 může nastat až po uskutečnění obou dvou událostí a3,a4.



OSTRAVSKÁ  
UNIVERZITA  
PŘÍRODOVĚDECKÁ FAKULTA

# Vyloučení souběhu činnosti

## Pravidelné střídání

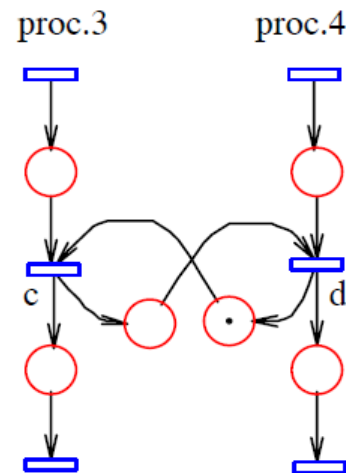


### Vyloučení souběhu

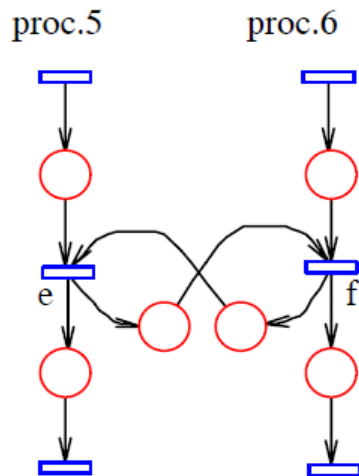
Události a, b procesů 1,2  
nemohou nastat současně

### Pravidelné střídání

Události c, d procesů  
3, 4 se musí  
pravidelně střídat



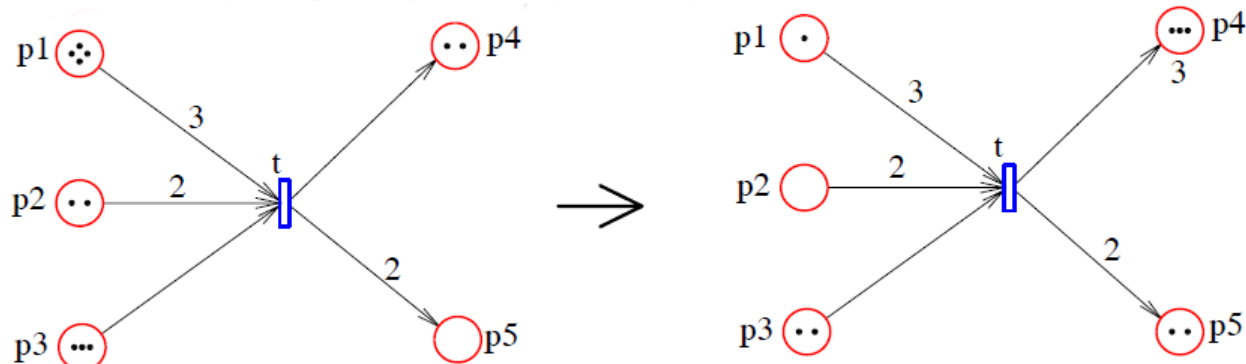
# Deadlock - uzamčení



Žádná z událostí e, f procesů 5, 6  
nemůže nastat.

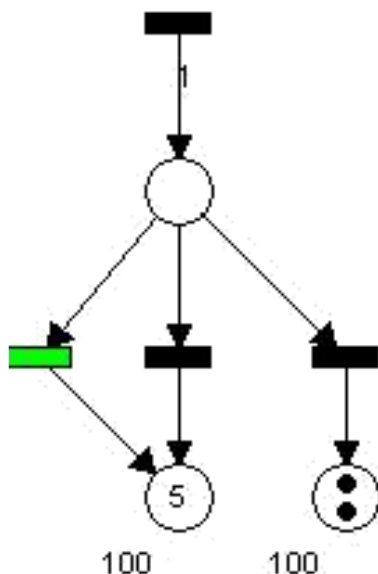
# Ohodnocení hran a přechodů

- Místo  $p$  je určeno kapacitou  $c(p)$  maximálního počtu značek.
- Přechod může být uskutečněn jen pokud (současně se splněním vstupních podmínek) není překročena kapacita výstupních podmínek.
- Počty odebíraných (umístěných) značek jsou specifikovány váhou hrany.
- Přechod je uskutečněn jen pokud jsou vstupní hrany nasyceny, tj. pokud není počet značek ve vstupních místech daného přechodu menší než váhy příslušných hran.



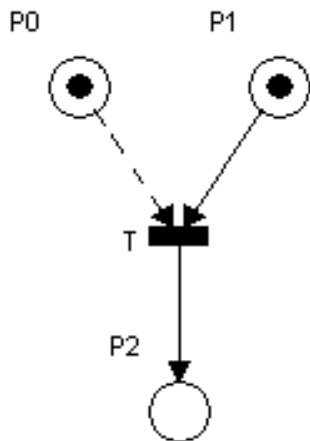


# Konfliktní přechody

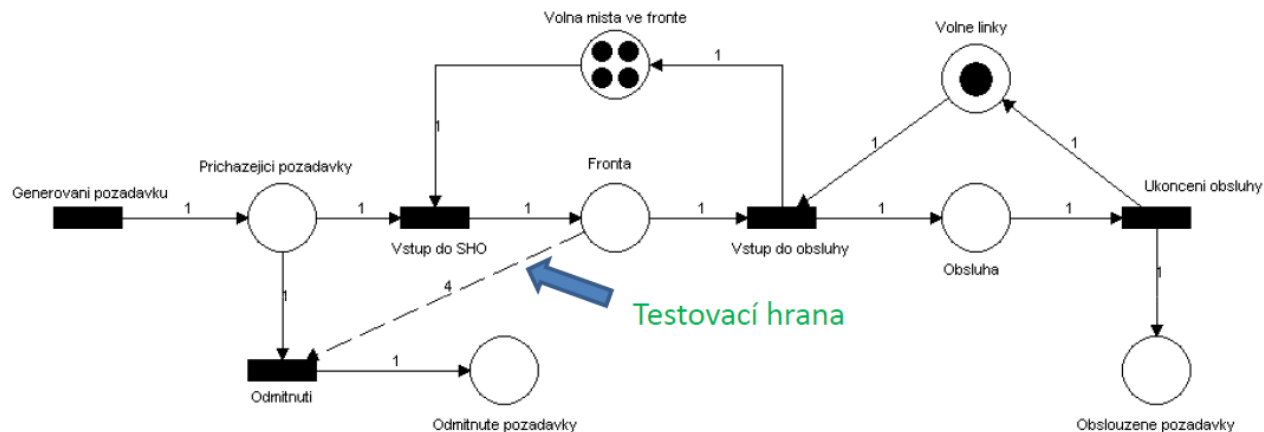


- Dva současně proveditelné přechody jsou konfliktní, když provedení jednoho způsobí, že druhý přestane být proveditelný.
- Konfliktní přechody modelují soupeření o zdroje a vzájemnou výlučnost dvou událostí.
- Nezávislé přechody modelují asynchronnost a paralelismus.

# Testovací hrany



- Testovací hranu používáme pro vstupní podmínky a to v případě že pro odpal (realizaci) přechodu testovaná podmínka musí být splněna, ale odpalem přechodu se platnost podmínky nezmění.
- Testovací hranu graficky znázorňujeme čárkovanou čarou. Podobně jako obyčejná hrana může být i testovací hrana ohodnocená.
- Pro to, aby byl přechod v aktuálním značení aktivní je nutné, aby počet žetonů v místě byl alespoň roven hodnotě testovací hrany.
- Odpálením přechodu se počet žetonů v místě, spojeném s přechodem testovací hranou nezmění.

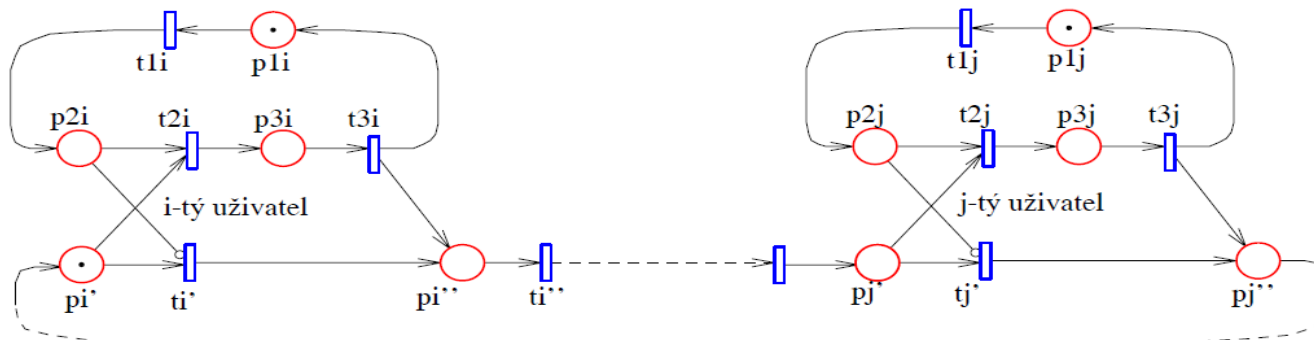


- Na obrázku je model systému hromadné obsluhy se 4 místy ve frontě a 1 obslužnou linkou vytvořený pomocí P/T Petriho sítě.
- V této síti je uvažována nekonečná kapacita všech míst.
- Proto bylo nutné pro zachování správné funkce modelovaného systému použít pomocná místa „Volná místa ve frontě“ a „Volné linky“ – tzv. komplementární místa.

# Inhibiční hrana

- Je speciální hrana, která může směřovat pouze z místa k přechodu.
- Místo šipky zakresluje kolečko.
- Inhibiční hrany upravují proveditelnost přechodů.
- Přechod  $t$  je proveditelný, jestliže:
  - Je přechod  $t$  proveditelný v příslušné P/T Petriho síti (tj. v síti, která vznikne z dané sítě odstraněním inhibičních hran).
  - Pro každé místo  $p$  vstupní inhibiční množiny přechodu  $t$  platí, že obsahuje méně tokenů než kolik činí násobnost inhibiční hrany vedoucí z místa  $p$  do přechodu  $t$ .
- Po inhibiční hraně se tokeny nepřesouvají



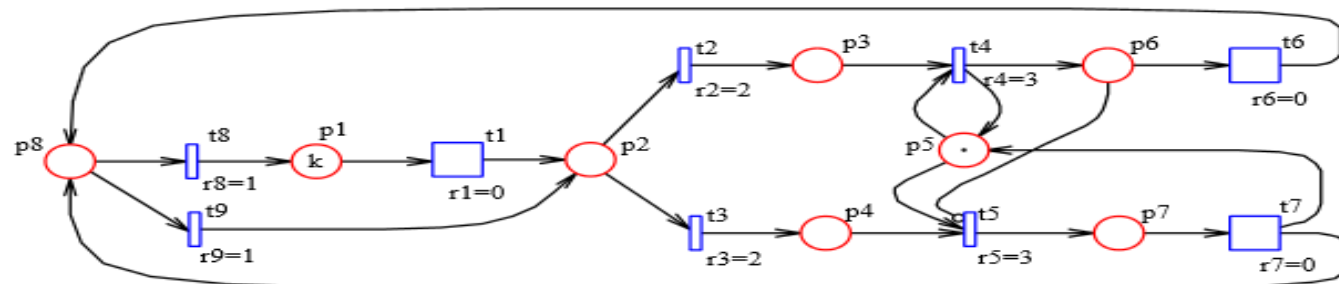


- $p1i$ : stav i-tého terminálu: bez zprávy k odeslání
- $t1i$ : vznik zprávy v i-tém terminálu
- $p2i$ : zpráva je vyhotovena a i-tý terminál čeká na příchod tokenu v síti
- $t2i$ : i-tý terminál přebírá kontrolu nad komunikací v síti
- $p3i$ : zpráva od i-tého terminálu je předávána v síti
- $t3i$ : zpráva byla předána, i-tý terminál vrací kontrolu
- $pi'$ : i-tý terminál je na řadě, má-li zprávu, může ji předat
- $ti'$ : i-tý terminál rezignuje na kontrolu sítě
- $pi''$ : i-tý terminál skončil kontrolu sítě
- $ti''$ : kontrola je předávána následujícímu  $(i+1)$ -ému terminálu

# Petriho síť s prioritami

- Petriho síť s prioritami je P/T Petriho síť, ve které je každému přechodu přiřazeno celé nezáporné číslo udávající prioritu přechodu. Priority přechodů upravují pravidla pro jejich provádění.
- V Petriho síti s prioritami je přechod  $t$  povolen, je-li proveditelný v odpovídající P/T síti bez priorit.
- Přechod  $t$  je proveditelný tehdy, pokud:
  - Je povolen.
  - Žádný jiný povolený přechod nemá vyšší prioritu.
- Po provedení přechodu  $t$  se značení sítě změní





- P1: značení udává počet procesů, které v daný okamžik nevyžadují přístup k databázi  
t1: vznik potřeby přístupu do databáze u některého z procesů  
p2: značení udává počet procesů, které v daný okamžik vyžadují přístup k databázi a to buď kvůli čtení nebo kvůli zápisu  
t2: požaduje se přístup do databáze za účelem čtení  
t3: požaduje se přístup do databáze za účelem zápisu  
p3: značení udává počet procesů, které v daný okamžik, čekají na čtení z databáze  
p4: značení udává počet procesů, které v daný okamžik, čekají na zápis do databáze  
p5: token značí, že neprobíhá zápis do databáze  
t4: začátek procesu čtení  
t5: začátek procesu zápisu  
p6: značení udává počet procesů čtoucích z databáze  
p7: token značí, že probíhá zápis do databáze  
t6: ukončení procesu čtení  
t7: ukončení procesu zápisu  
p8: značení udává počet procesů, které ukončily přístup do databáze  
t8: proces bude po určitou dobu pracovat pouze s lokální pamětí  
t9: proces vyžaduje okamžitě další přístup do společné databáze

# Časované Petriho sítě

- Trvání dějů může být:
  - Deterministické – v tomto případě se hovoří o Časovaných Petriho sítích.
  - Stochastické – v tomto případě hovoříme o Stochastických Petriho sítích (SPN – Stochastic Petri Nets).
  - Kombinované – v tomto případě hovoříme o Zobecněných stochastických Petriho sítích (GSPN – Generalised Stochastic Petri Nets).





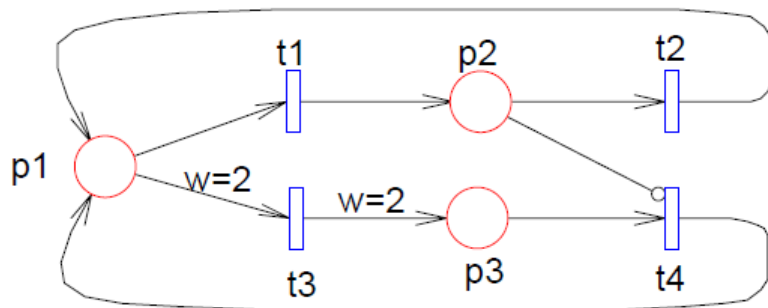
# Časované Petriho sítě

Zavedení času může být spojeno s:

- **Přechody** (T-timed PN) – provedení přechodu trvá určitou dobu, po kterou token pobývá uvnitř přechodu.
- **Místy** (P-timed PN) – token pobývá stanovenou dobu ve vstupním místě přechodu, jež má být proveden.
- **Hranami** (A-timed PN) – přesun tokenu po příslušné hraně trvá určitou dobu.
- **Tokeny** (Token Timed PN) – Provádění přechodů v síti je sice okamžité, ale tokeny opouštějící příslušný přechod jsou opatřeny **časovým razítkem** (time stamp), jež udává, kdy může být daný token zase použit. Hodnota časového razítka odpovídá aktuální hodnotě simulárního času zvětšnou o příslušnou hodnotu.



# Maticová reprezentace PN



- Vstupní matice – udává kolik značek ubude pro dané místo a přechod.
- Výstupní matice – udává kolik značek přibude pro dané místo a přechod
- Toková matice – zachycuje vstupní a výstupní matici
- Matice změn (incidenční matice) – reprezentuje změnu pro dané místo a přechod

$$C = O^T - I^T = \begin{array}{cc|cc|cc|cc} & t_1 & t_2 & t_3 & t_4 & & & & \\ \hline & 0 & 1 & 0 & 1 & 1 & 0 & 2 & 0 & p_1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & p_2 \\ 0 & 0 & 2 & 0 & 0 & 0 & 0 & 1 & & p_3 \end{array} = \begin{array}{cc|cc} t_1 & t_2 & t_3 & t_4 \\ \hline -1 & 1 & -2 & 1 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & 2 & -1 \end{array}$$