

Обојене Петријеве мреже

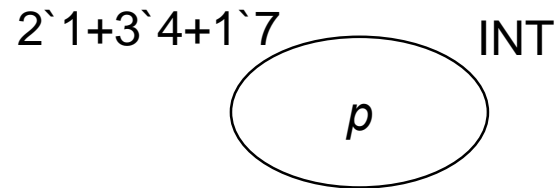
Обојене Петријеве мреже (ОПМ)

- *Colored Petri Nets* (CPN)
- Курт Јенсен (*Kurt Jensen*) 1981. године
- Сваком чвору типа место придружен је тип податка који одређује врсту податка које дато место може да садржи.
- Сваки жетон у месту носи вредност податка која припада типу тог места.
- Вредност податка се назива и боја жетона а тип податка – скуп боја.

Место ОПМ

- Мултискуп $M=(S, m)$, над непразним скупом S , где се функција $m:S \rightarrow \mathbf{N}$ се представља као формална сума $\sum_{s \in S} m(s) \cdot s$
- $m(s)$ представља број појављивања елемента $s \in S$ у мултскупу M .
- Маркирање у обојеним ПМ може да се представи као функција која сваком месту p придружује мултискуп жетона одговарајућег типа.

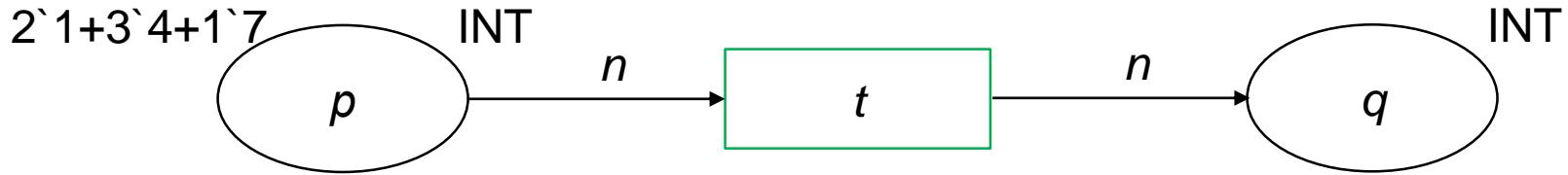
Маркирање места p које је типа
цео број је: 1, 1, 4, 4, 4 и 7.



Грана ОПМ

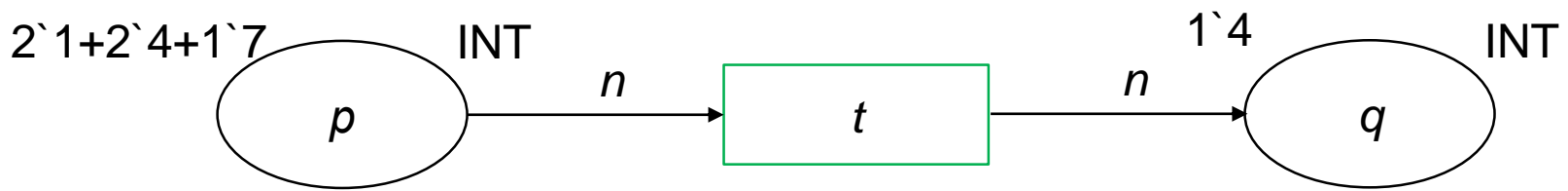
- Тежина гране - опис гране (*arc expression*).
- Опис гране одређује тачан број и врсту жетона која се паљењем прелаза t ослобађа из $p \in {}^0t$ и производи у $p \in t^0$.
- Опис гране мора бити истог типа као место са којим је повезана.
- Опис гране може бити променљива, константа или функција.

n – променљива типа *INT*



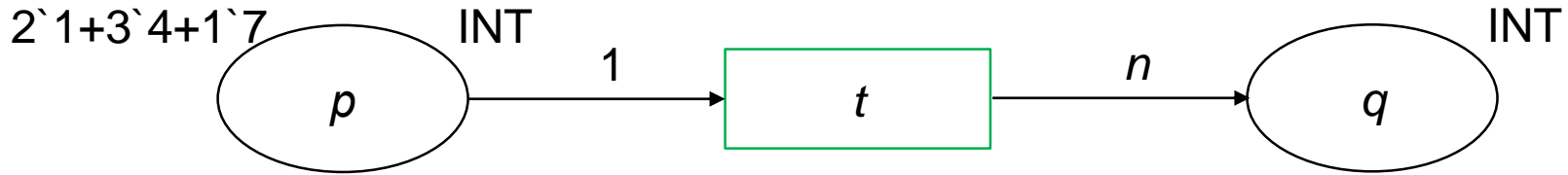
Грана ОПМ

- Тежина гране - опис гране (*arc expression*).
- Опис гране одређује тачан број и врсту жетона која се паљењем прелаза t ослобађа из $p \in {}^0t$ и производи у $p \in t^0$.
- Опис гране мора бити истог типа као место са којим је повезана.
- Опис гране може бити променљива, константа или функција.



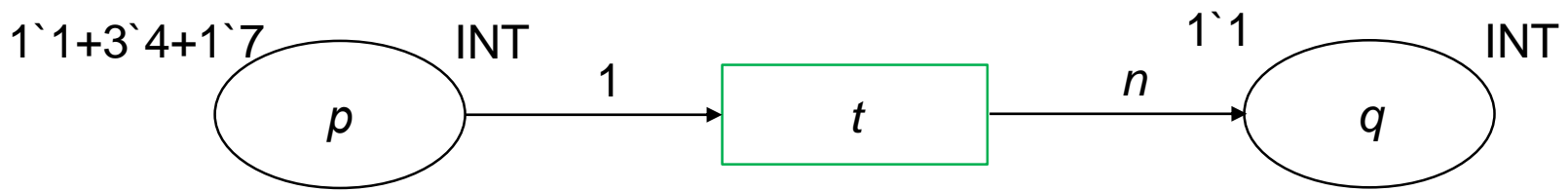
Грана ОПМ

- Тежина гране - опис гране (*arc expression*).
- Опис гране одређује тачан број и врсту жетона која се паљењем прелаза t ослобађа из $p \in t^o$ и производи у $p \in t^o$.
- Опис гране мора бити истог типа као место са којим је повезана.
- Опис гране може бити променљива, константа или функција.



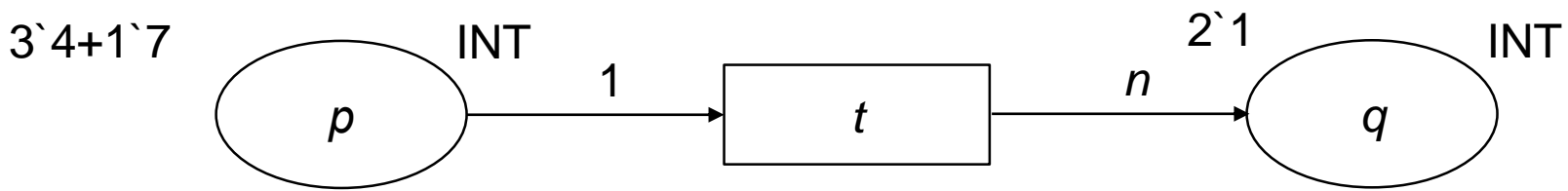
Грана ОПМ

- Тежина гране - опис гране (*arc expression*).
- Опис гране одређује тачан број и врсту жетона која се паљењем прелаза t ослобађа из $p \in {}^0t$ и производи у $p \in t^0$.
- Опис гране мора бити истог типа као место са којим је повезана.
- Опис гране може бити променљива, константа или функција.



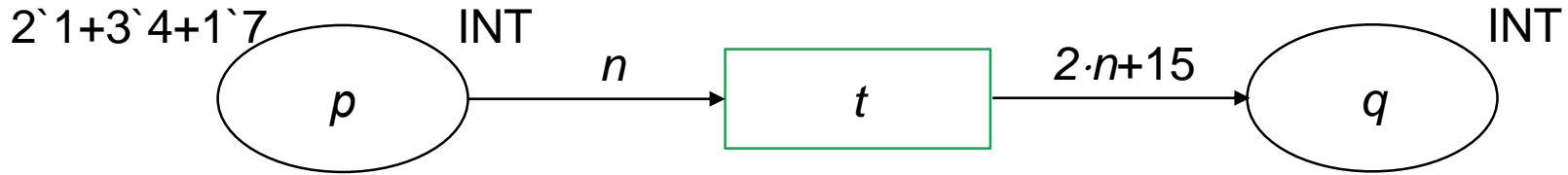
Грана ОПМ

- Тежина гране - опис гране (*arc expression*).
- Опис гране одређује тачан број и врсту жетона која се паљењем прелаза t ослобађа из $p \in {}^0t$ и производи у $p \in t^0$.
- Опис гране мора бити истог типа као место са којим је повезана.
- Опис гране може бити променљива, константа или функција.



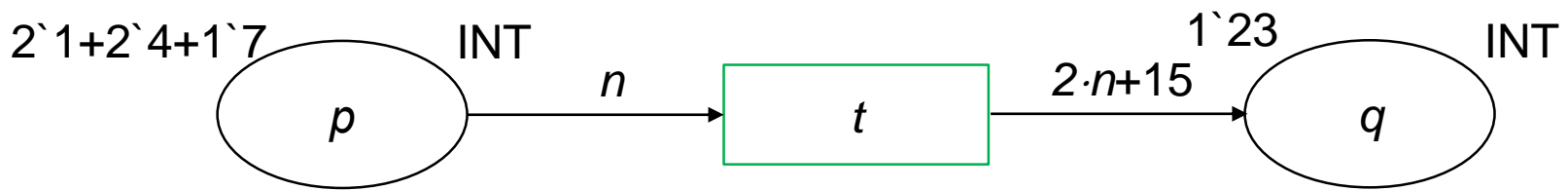
Грана ОПМ

- Тежина гране - опис гране (*arc expression*).
- Опис гране одређује тачан број и врсту жетона која се паљењем прелаза t ослобађа из $p \in {}^0t$ и производи у $p \in t^0$.
- Опис гране мора бити истог типа као место са којим је повезана.
- Опис гране може бити променљива, константа или функција.



Грана ОПМ

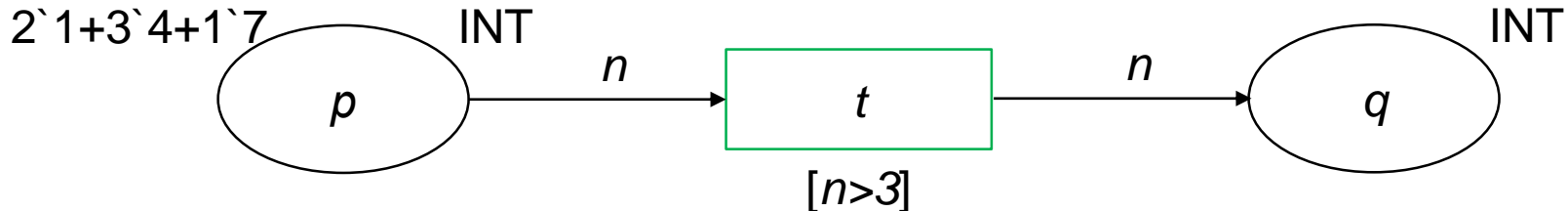
- Тежина гране - опис гране (*arc expression*).
- Опис гране одређује тачан број и врсту жетона која се паљењем прелаза t ослобађа из $p \in {}^0t$ и производи у $p \in t^0$.
- Опис гране мора бити истог типа као место са којим је повезана.
- Опис гране може бити променљива, константа или функција.



Прелаз ОПМ

- Паљење прелаза – *transition occurrence*.
- Прелазу се може доделити *guard* функција која одређује додатни услов паљења прелаза.
- Резултат *guard* функције је буловска променљива ($\{true, false\}$) а аргументи морају бити истог типа (боје) као места са којим је прелаз повезан.

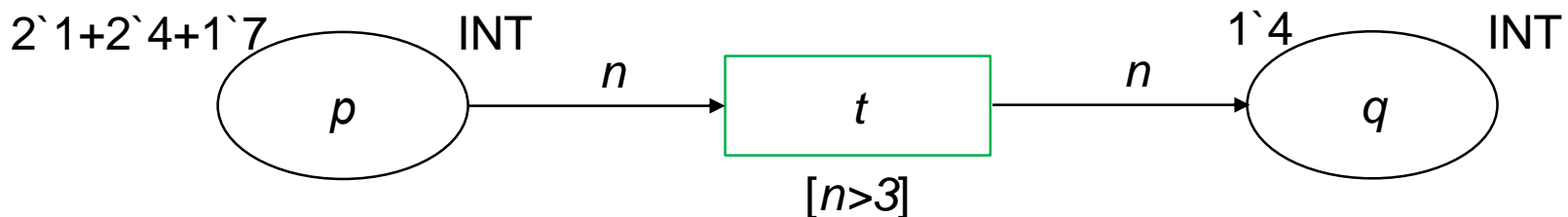
$$\forall t \in T : [\text{Type}(G(t)) = \mathbf{B} \wedge \text{Type}(\text{Var}(G(t))) \subseteq \Sigma]$$



Прелаз ОПМ

- Паљење прелаза – *transition occurrence*.
- Прелазу се може доделити *guard* функција која одређује додатни услов паљења прелаза.
- Резултат *guard* функције је буловска променљива ($\{true, false\}$) а аргументи морају бити истог типа (боје) као места са којим је прелаз повезан.

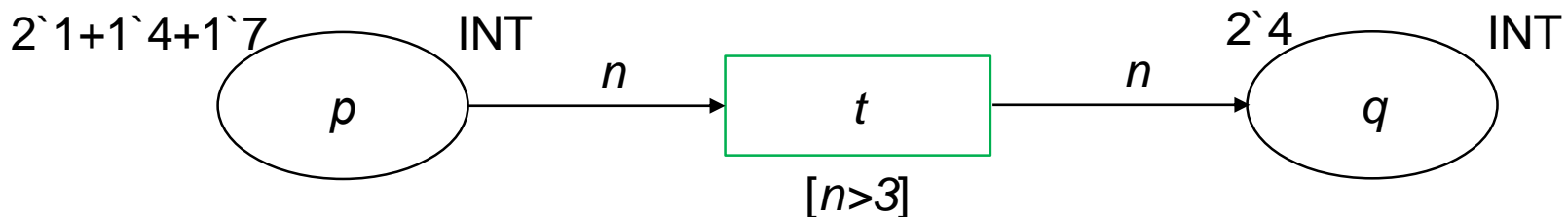
$$\forall t \in T : [\text{Type}(G(t)) = \mathbf{B} \wedge \text{Type}(\text{Var}(G(t))) \subseteq \Sigma]$$



Прелаз ОПМ

- Паљење прелаза – *transition occurrence*.
- Прелазу се може доделити *guard* функција која одређује додатни услов паљења прелаза.
- Резултат *guard* функције је буловска променљива ($\{true, false\}$) а аргументи морају бити истог типа (боје) као места са којим је прелаз повезан.

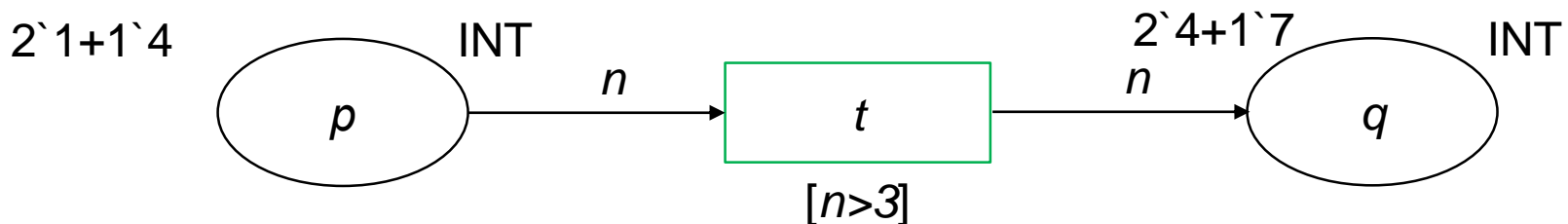
$$\forall t \in T : [\text{Type}(G(t)) = \mathbf{B} \wedge \text{Type}(\text{Var}(G(t))) \subseteq \Sigma]$$



Прелаз ОПМ

- Паљење прелаза – *transition occurrence*.
- Прелазу се може доделити *guard* функција која одређује додатни услов паљења прелаза.
- Резултат *guard* функције је буловска променљива ($\{true, false\}$) а аргументи морају бити истог типа (боје) као места са којим је прелаз повезан.

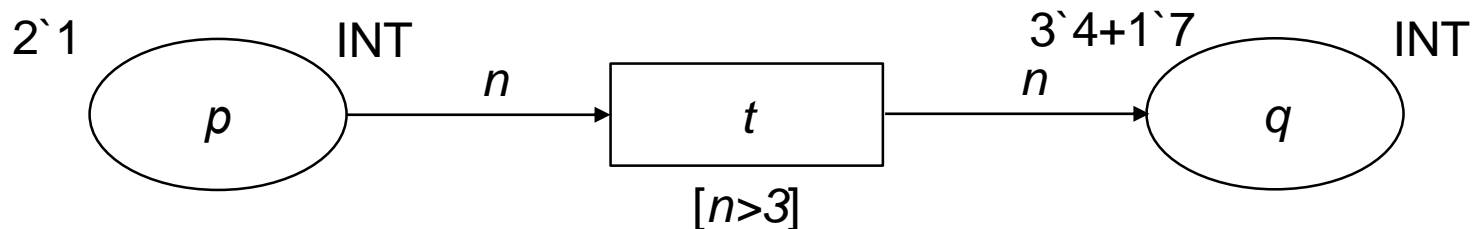
$$\forall t \in T : [\text{Type}(G(t)) = \mathbf{B} \wedge \text{Type}(\text{Var}(G(t))) \subseteq \Sigma]$$



Прелаз ОПМ

- Паљење прелаза – *transition occurrence*.
- Прелазу се може доделити *guard* функција која одређује додатни услов паљења прелаза.
- Резултат *guard* функције је буловска променљива ($\{true, false\}$) а аргументи морају бити истог типа (боје) као места са којим је прелаз повезан.

$$\forall t \in T : [\text{Type}(G(t)) = \mathbf{B} \wedge \text{Type}(\text{Var}(G(t))) \subseteq \Sigma]$$



Формална дефиниција CPN

Обојена PN је деветорка $CPN = (\Sigma, P, T, A, N, C, G, E, M_0)$ где је:

Σ	коначан скуп типова (боја)
$P = \{p_1, p_2, \dots, p_n\}$	коначан скуп места
$T = \{t_1, t_2, \dots, t_q\}$	коначан скуп прелаза
A	коначан скуп грана такав да $P \cap T = P \cap A = T \cap A = \emptyset$
$N: A \rightarrow (P \times T) \cup (T \times P)$	функција чворова
$C: P \rightarrow \Sigma$	функција типова (боја)

G функција чувања, која пресликава T у израз такав да:

$$t \in T: [\text{Type}(G(t)) = B \wedge \text{Type}(\text{Var}(G(t))) \subseteq \Sigma]$$

E опис гране, који пресликава A у израз такав да:

$$\forall a \in A: [\text{Type}(E(a)) = C(p)_{MS} \wedge \text{Type}(\text{Var}(E(a))) \subseteq \Sigma]$$

M_0 почетна функција, која пресликава P у затворени израз такав да:

$$\forall p \in P: [\text{Type}(M_0(p)) = C(p)_{MS}]$$

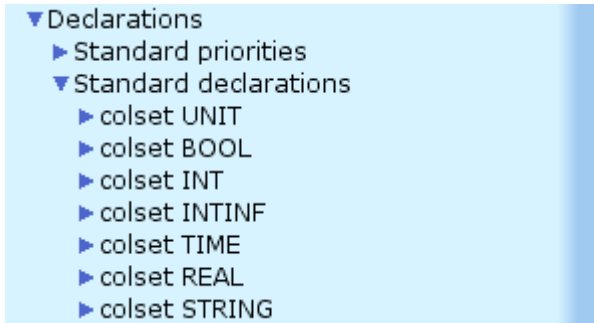
Обојена Петријевих мрежа

Обојена Петријева мрежа којом се моделира конкретан процес се састоји из 3 дела:

- Структура мреже, односно оријентисан, бипартитни граф дефинисан са $(ii -v)$;
- Опис мреже (*declaration*);
- Ознаке на мрежи: типови места, описи грана, услови паљења прелаза и почетно маркирање на мрежи.

CPN Tools

- Опис мреже (*declaration*);

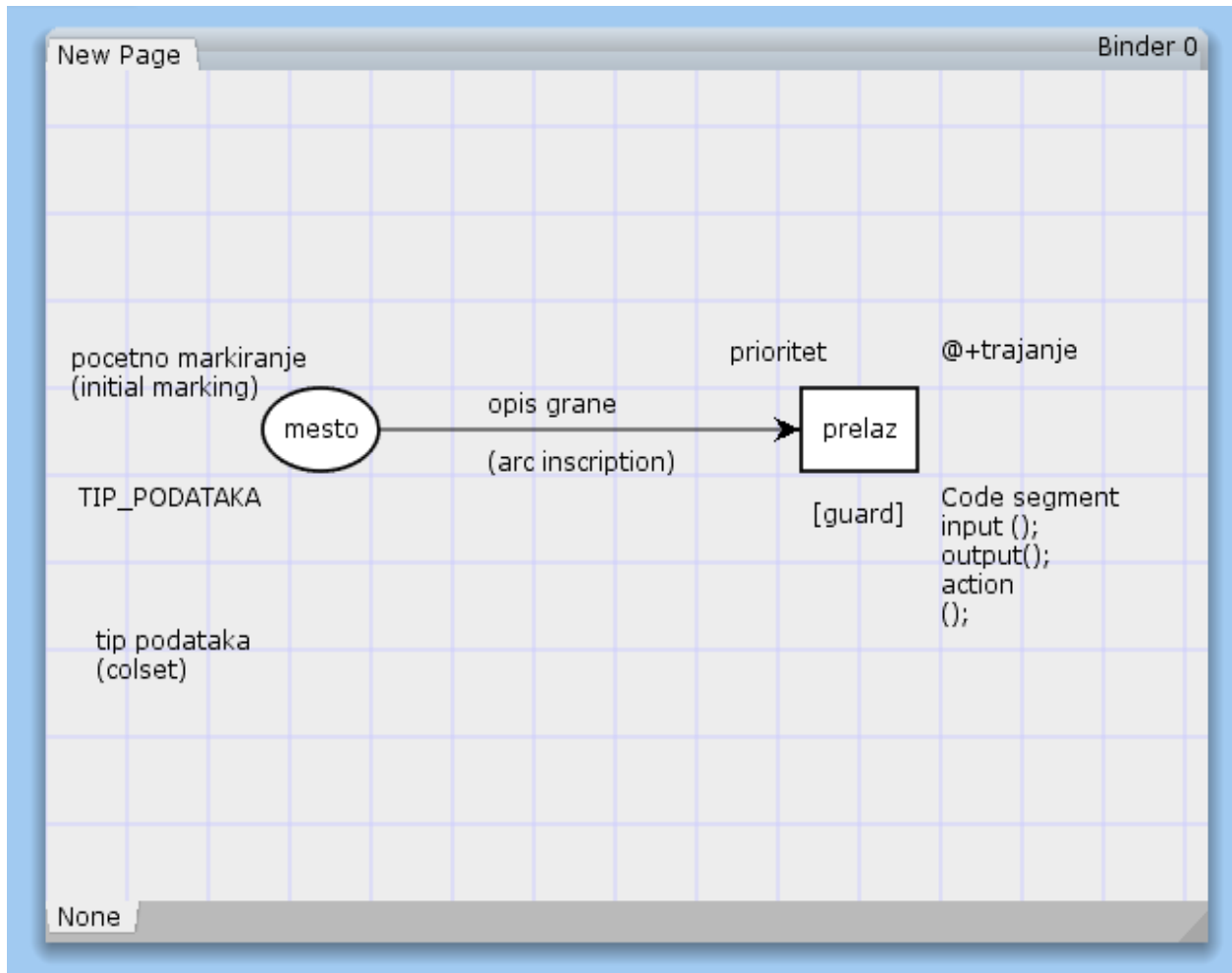


- Декларишу се:

- Типови података (colset),
- Функције (fun),
- Променљиве (var),
- Константе - параметри (val).

CPN Tools

- Ознаке на мрежи (*inscription*): типови места, описи грана, услови паљења прелаза и почетно маркирање на мрежи.



CPN Tools - Основни типови података (*Simple color sets*)

colset UNIT = unit; неструктурирани тип података, вредност токена = ()

синтакса: colset name = unit [with new_unit];

пример: colset E = unit with e;

colset BOOL = bool; буловски тип података, вредност токена $\in \{\text{true}, \text{false}\}$

синтакса: colset name = bool [with (new_false, new_true)];

пример: colset Uslov= bool with (da, ne);

var a: Uslov;

CPN Tools - Основни типови података (*Simple color sets*)

colset INT = int; целобројни тип података, вредност токена $\in \mathbf{Z}$

синтакса: colset name = int [with int-exp1..int-exp2];

пример: colset Broj =int;
colset Broj =int with 1..5;
var n : INT;
var n : Broj;

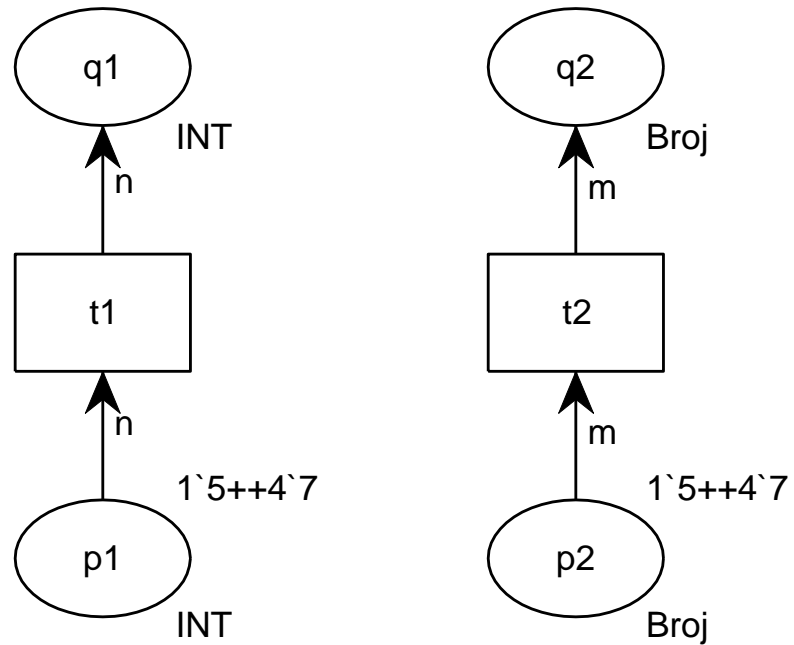
colset INTINF = intinf; целобројни тип података, вредност токена $\in \mathbf{Z}$ и $\leq \infty$

синтакса: colset name = intinf [with int-exp1..int-exp2];

пример: colset Velikibroj =intinf;
colset Velikibroj =intinf with 1..500000;
var n : INTINF;
var n : Velikibroj;

(* Standard declarations *)

```
colset INT = int;  
colset Broj = int;  
var n:INT;  
var m:Broj;
```



Тип података Broj је тзв. *Alias color sets* – тип података који има исте вредности и својства као неки претходно дефинисани тип података. Сврха његовог постојања је прецизније описивање неког типа података и лакша чилџивост ПМ.

Пример

Испред мењачнице се налази 10 људи од којих неки хоће да замене динаре за евре (купе €) а неки да замене евре за динаре (продају €).

особа	A	B	C	D	E	F	G	H	I	J
купује €	100			500		200	100			300
продаје €		200	300		100			400	200	

У каси мењачнице се тренутно налази 500 € а динара има довољно.

Клијенти се услужују на случајан начин.

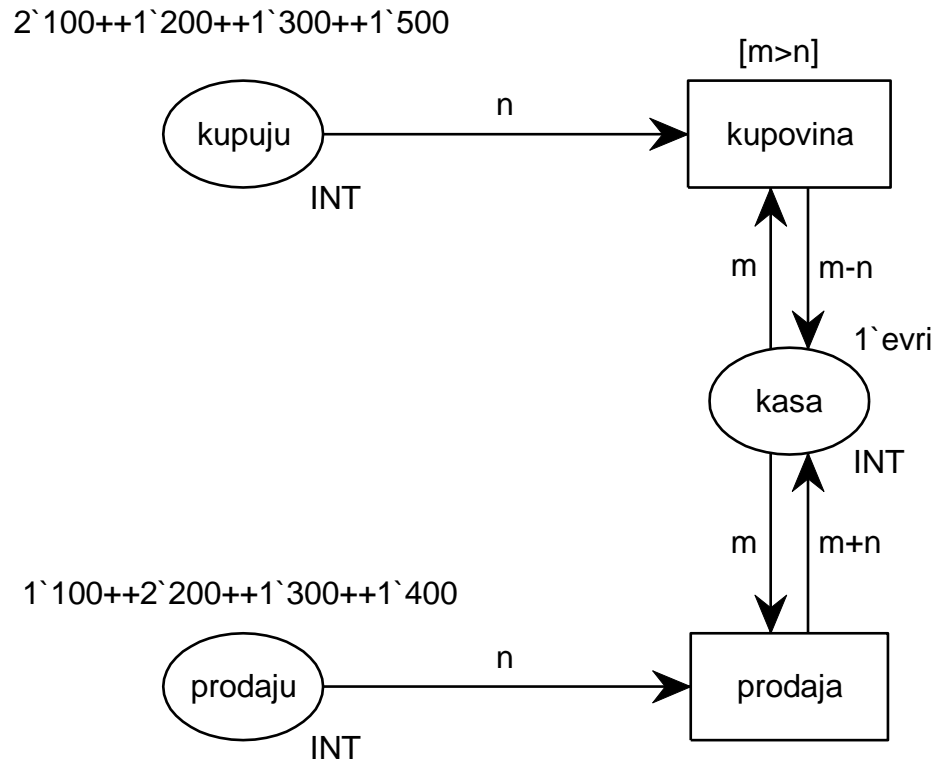
Моделирати помоћу ОПМ и симулирати добијену мрежу.

(* Standard declarations *)

colset INT = int;

var n, m: INT;

val evri = 500;



CPN Tools - Основни типови података (*Simple color sets*)

colset REAL= real; реални тип података, вредност токена $\in \mathbf{R}$

синтакса: colset name = real [with real-exp1..real-exp2];

пример: colset Broj =real;

colset Broj =real with 1.3..5.1;

var *n*: REAL;

var *n*: Broj;

colset STRING= string; текстуални тип података

синтакса: colset name = string [with string-exp1..string-exp2 [and int-exp1..int-exp2]];

пример: colset Tekst =string;

colset Tekst =string with "a".."k";

colset Tekst =string with "a".."k" and 5..8;

var *a*: STRING;

var *a*: Tekst;

CPN Tools - Основни типови података (*Simple color sets*)

colset TIME = time; вредност токена зависи од системског времена

синтакса: colset name = time;

пример: colset Vreme =time;

var t: TIME;

var t: Vreme;

time() опис гране – резулат је системско време

Enumeration color set набројиви тип података

синтакса: colset name = with id0 | id1 | ... | idn;

пример: colset Dani =with pon | ut | sre | cet | pet | sub | ned;

colset Svrha =with uplata | isplata;

var a: Dani;

var a: Svrha;

CPN Tools – Сложени типови података (*Compound color sets*)

Product color sets

уређене n -торке више типова података

синтакса: `colset name = product name1 * name2 * ... * namen;`

пример: `colset Klijent = product Svrha * Iznos;`

`var k: Klijent;` вредност токена нпр. (uplata, 100)

`var s: Svrha;`

`var n: Iznos;`

Опис гране: k или (s,n)

Пример

Испред мењачнице се налази 10 људи од којих неки хоће да замене динаре за евре (купе €) а неки да замене евре за динаре (продају €).

особа	A	B	C	D	E	F	G	H	I	J
купује €	100			500		200	100			300
продаје €		200	300		100			400	200	

У каси мењачнице се тренутно налази 500 € а динара има довољно.

Клијенти се услужују на случајан начин.

Моделирати помоћу ОПМ тако да постоји једно место чије почетно маркирање чине сви клијенти и симулирати добијену мрежу.

(* Standard declarations *)

colset INT = int;

colset Svrha = with kupuje | prodaje;

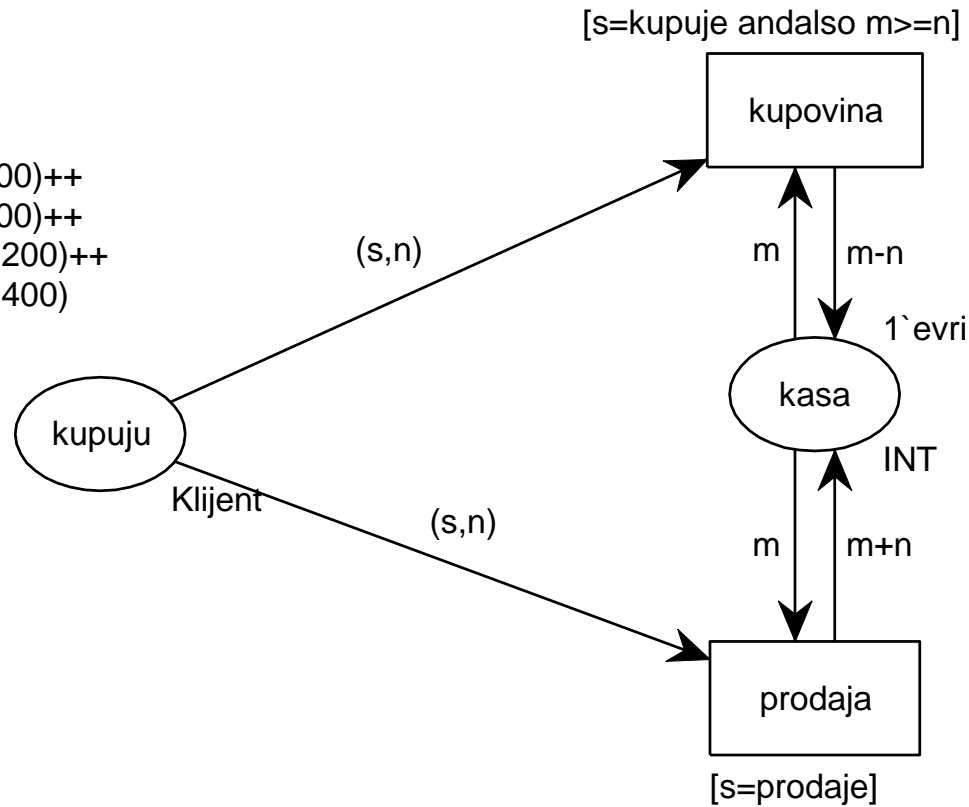
colset Klijent = product Svrha * INT;

var n, m: INT;

var s: Svrha;

val evri = 500;

2` (kupuje,100)++1` (kupuje,200)++
1` (kupuje,300)++1` (kupuje,500)++
1` (prodaje,100)++2` (prodaje,200)++
1` (prodaje,300)++1` (prodaje,400)



Пример

Испред мењачнице се налази 10 људи од којих неки хоће да замене динаре за евре (купе €) а неки да замене евре за динаре (продају €).

особа	A	B	C	D	E	F	G	H	I	J
купује €	100			500		200	100			300
продаје €		200	300		100			400	200	

У каси мењачнице се тренутно налази 500 € и 70000 динара. Продајни курс износи 122,49 динара а куповни 123,23 за један €.

Клијенти се услужују на случајан начин.

Моделирати помоћу ОПМ тако да постоји једно место чије почетно маркирање чине сви клијенти и симулирати добијену мрежу.