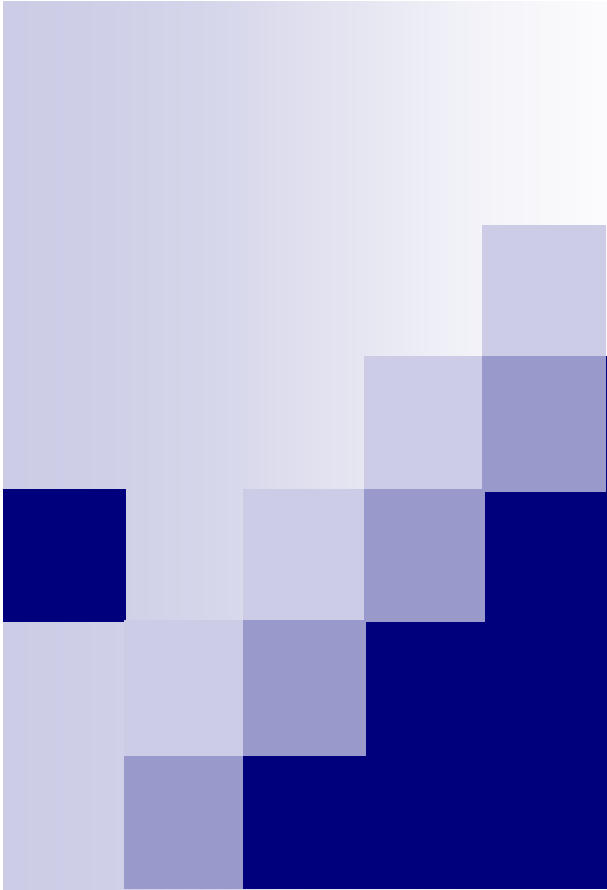




Notițe de curs - 6 - Gestionarea șirurilor în Arena

Radu VLAD

*Modelarea și simularea sistemelor
Discrete-Event System Simulation*

Precizarea caracteristicilor șirurilor

- Definirea șirurilor – modulul Queue

The screenshot displays the Arena software interface for a model titled "[Jonctiune]". The left sidebar shows the "Basic Process" tree with the "Queue" module selected. The main workspace contains a flow diagram with three input boxes: "Genereaza trenuri pentru traseul 1", "Genereaza trenuri pentru traseul 2", and "Animation". Arrows from the first two boxes point to a box labeled "Modelul jonctiunii". Below the diagram is a table titled "Queue - Basic Process" with the following data:

	Name	Type	Shared	Report Statistics
1	Q_J1_L1_Departure	First In First Out	☐	☒
2	Q_J1_L2_Departure	First In First Out	☐	☒
3	Q_J1_L1_Signal1	First In First Out	☐	☒

Double-click here to add a new row.

Precizarea caracteristicilor șirurilor

■ Dezavantajele modului Queue

- Imposibilitatea:
 - impunerii unei limite lungimii șirurilor de așteptare,
 - modelării "găurilor neagre"

■ Blocul Queue

- Capacity
 - limitează numărul de entități care pot aștepta simultan,
- Balk Label
 - indică blocurile spre care vor fi direcționate entitățile care găsesc șirul plin,
- Detach
 - opțiune care împiedică trecerea entităților spre modulul/blocul următor.

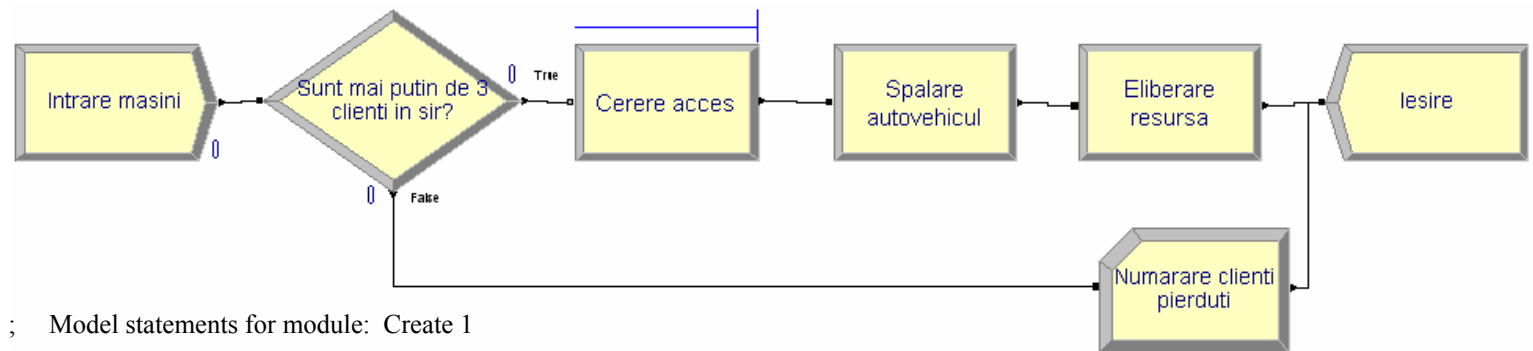
The screenshot shows a 'Queue Block' dialog box. It has a blue title bar with a question mark icon and a close button. The main area is light beige. It contains the following elements:

- 'Label': A text input field.
- 'Mark Attribute': A dropdown menu.
- 'Queue ID': A dropdown menu.
- 'Capacity': A text input field.
- 'Balk Label': A text input field.
- 'Blockages...': A button.
- 'Detach': A checkbox.
- 'Comments': A text area.
- 'OK', 'Cancel', and 'Help': Buttons at the bottom.

Utilizarea blocului Queue

Simularea activității unei spălătorii auto

- Managerul stației a propus modificarea anterioară deoarece a observat că în varianta existentă de organizare a activității pierde clienți atunci când în șir se găsesc mai mult de trei mașini. (lucrare laborator 3)



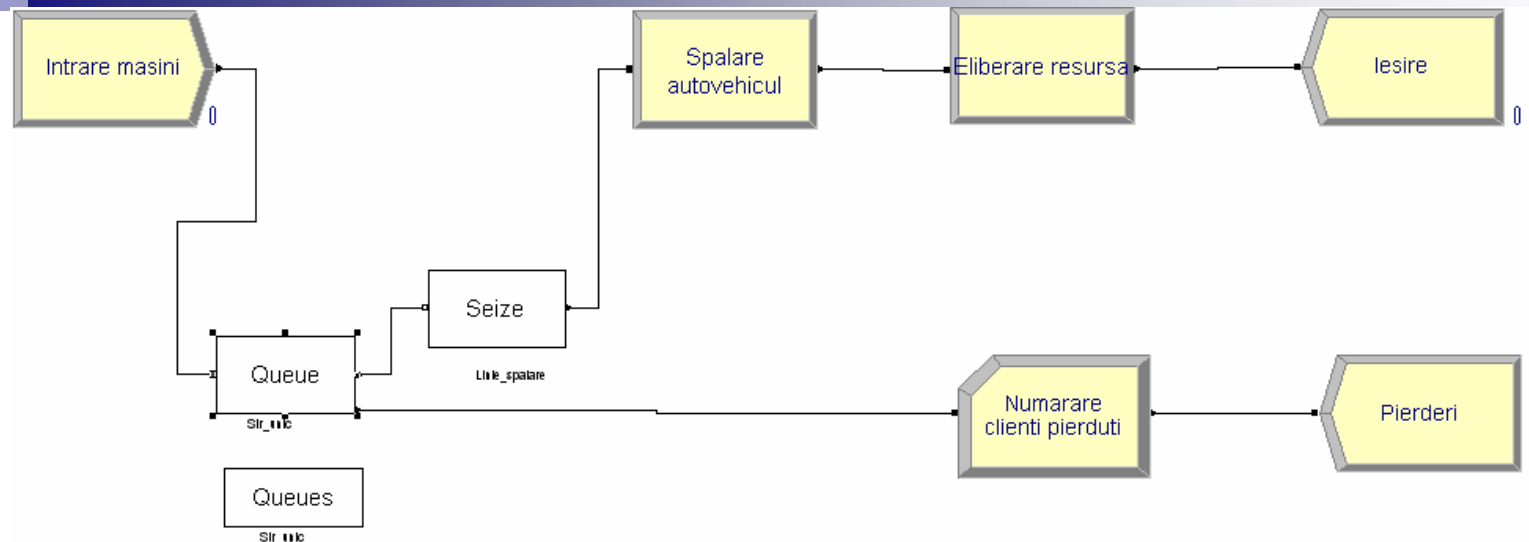
```

; Model statements for module: Create 1
7$ CREATE, 1,MinutesToBaseTime(0.0),Entity 1: MinutesToBaseTime(EXPO(5));
8$ ASSIGN: Intrare masini.NumberOut=Intrare masini. NumberOut + 1:NEXT(5$);
; Model statements for module: Decide 1
5$ BRANCH, 1:
    If,NQ(Sir_unic)<3,11$,Yes:
        Else,12$,Yes;
11$ ASSIGN: Sunt mai putin de 3 clienti in sir?.NumberOut True = Sunt mai putin de 3
    clienti in sir?.NumberOut True + 1:NEXT(0$);
12$ ASSIGN: Sunt mai putin de 3 clienti in sir?.NumberOut False = Sunt mai putin de 3
    clienti in sir?.NumberOut False + 1:NEXT(6$);
; Model statements for module: Seize 1
0$ QUEUE, Sir_unic;
    SEIZE, 2,Other: Linie_spalare,1:NEXT(14$);
14$ DELAY: 0.0,,VA:NEXT(2$);
  
```

```

; Model statements for module: Delay 1
2$ DELAY: MinutesToBaseTime(NORM( 15, 3)):NEXT(3$);
; Model statements for module: Release 1
3$ RELEASE: Linie_spalare,1:NEXT(4$);
; Model statements for module: Dispose 1
4$ ASSIGN: Iesire.NumberOut=Iesire.NumberOut + 1;
15$ DISPOSE: Yes;
; Model statements for module: Record 1
6$ COUNT: Clienti_pierduti,1:NEXT(4$);
  
```

Utilizarea blocului Queue



```

; Model statements for module: Create 1
7$ CREATE, 1,MinutesToBaseTime(0.0),Entity 1:
    MinutesToBaseTime(EXPO(5)): NEXT(8$);
8$ ASSIGN: Intrare masini.NumberOut=
    Intrare masini.NumberOut + 1:NEXT(3$);

3$ QUEUE, Sir_unic,3,5$;
4$ SEIZE, 1,Other: Linie_spalare,1:NEXT(0$);

; Model statements for module: Delay 1
0$ DELAY: MinutesToBaseTime(NORM( 15, 3)),Other:NEXT(1$);

; Model statements for module: Release 1
1$ RELEASE: Linie_spalare,1:NEXT(2$);
    
```

```

; Model statements for module: Dispose 1
2$ ASSIGN: Iesire.NumberOut=Iesire.NumberOut + 1;
11$ DISPOSE: Yes;

; Model statements for module: Record 1
5$ COUNT: Clienti_pierduti,1:NEXT(6$);

; Model statements for module: Dispose 2
6$ ASSIGN: Pierderi.NumberOut=Pierderi.NumberOut + 1;
12$ DISPOSE: Yes;
    
```

Extragerea entităților din șiruri

■ Blocul Search

- rol - determinarea indexului poziției ocupate de prima entitate ale cărei atribute îndeplinesc o anumită condiție.
- Search Item – indică șirul în care va fi cautată entitatea.
- Starting Index, Ending Index – blocul Search investighează numai entitățile de pe pozițiile precizate de cei doi indici.
- Search Condition – permite precizarea condiției pe care trebuie să o satisfacă entitățile.
- Atunci când găsește o entitate care satisface condiția specificată blocul înscrie indicele entității în variabila J.
- Dacă nu identifică nici o entitate, blocul înscrie în J valoarea 0.

Search Block

Label:

Mark Attribute:

Next Label:

Search Item:

Starting Index:

Ending Index:

Search Condition:

Comments:

OK Cancel Help

Extragerea entităților din șiruri

■ Blocul Remove

- ☐ rol - extragerea unei entități și trimiterea ei la o anumită adresă.
- ☐ nu trebuie lansat în execuție dacă J are valoarea 0.
- ☐ entitatea care a activat blocul Remove își continuă parcursul fără întârziere.

■ Exemplu

- ☐ Clienții care utilizează spălătoria auto (lucrarea de laborator 3) se plictisesc și părăsesc stația dacă așteaptă mai mult de 2 minute.

Remove Block

Label:

Mark Attribute:

Next Label:

Rank of Entity:

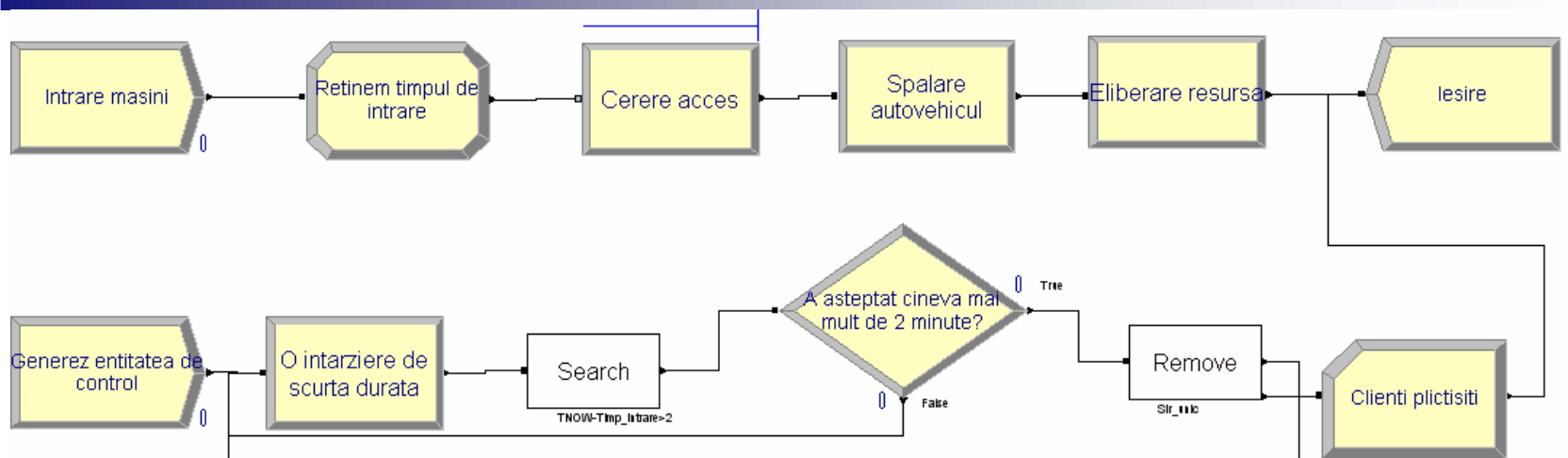
Queue ID:

Removed Entity Destination:

Comments:

OK Cancel Help

Extragerea entităților din șiruri



```

; Model statements for module: Create 2
18$ CREATE, 1,HoursToBaseTime(0.0),Entity
    1:HoursToBaseTime(EXPO(1)),1:NEXT(19$);
19$ ASSIGN: Generez entitatea de control.NumberOut=Generez
    entitatea de control.NumberOut + 1:NEXT(9$);

; Model statements for module: Delay 3
9$ DELAY: 1,,Other:NEXT(7$);

7$ SEARCH, Sir_unic,1,NQ:TNOW-Timp_intrare>2:NEXT(8$);

; Model statements for module: Decide 1
8$ BRANCH, 1:
    If,J.GT.0,22$,Yes:
    Else,23$,Yes;
  
```

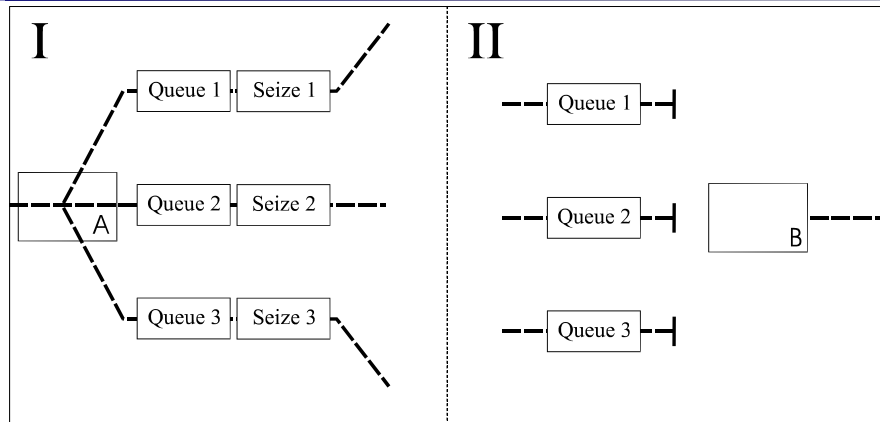
```

22$ ASSIGN: A asteptat cineva mai mult de 2 minute?.NumberOut
    True=A asteptat cineva mai mult de 2 minute?. NumberOut
    True + 1:NEXT(6$);
23$ ASSIGN: A asteptat cineva mai mult de 2 minute?.NumberOut
    False=A asteptat cineva mai mult de 2 minute?.NumberOut
    False + 1:NEXT(9$);

6$ REMOVE: j,Sir_unic,10$:NEXT(9$);

; Model statements for module: Record 1
10$ COUNT: Clienti_plictisiti,1:NEXT(4$);
  
```


Selecția șirurilor



■ Modulul PickQ

- Rol – selecția șirului spre care va fi direcționată entitatea activă.

The screenshot shows the 'PickQ Block' configuration window. It has a title bar with a question mark and a close button. The main area contains several fields: 'Label' (empty), 'Queue Selection Rule' (set to 'POR'), 'Balk Label' (empty), and 'Queue Labels' (a list box containing 'Linia_de_spalare_1', 'Linia_de_spalare_2', 'Linia_de_spalare_3', and '<End of list>'). To the right of the list box are buttons for 'Add...', 'Edit...', and 'Delete'. At the bottom is a 'Comments' text area and three buttons: 'OK', 'Cancel', and 'Help'.

Parametri:

Queue Selection Rule - precizează regula după care are loc selecția șirurilor de așteptare,

Queue Labels - conține lista etichetelor asociate blocurilor Queue la care poate fi trimisă entitatea activă,

Balk Label - indică adresa blocului sau a modului la care va fi trimisă entitatea activă dacă toate șirurile indicate de lista din *Queue Labels* nu mai au poziții libere.

Selecția șirurilor

■ Modulul QPick

- blocul *QPick* se găsește din punct de vedere logic între șirurile de așteptare și restul modelului.
- blocurile *Queue* asociate au selectat modificatorul *Detach*,

■ Criterii de selecție a șirului:

CYC - este selectat primul șir care urmează șirului selectat anterior,

LNQ - entitatea activă va fi trimisă spre șirul cu cele mai multe poziții ocupate,

LRC - determină alegerea șirului cu numărul cel mai mare de poziții libere,

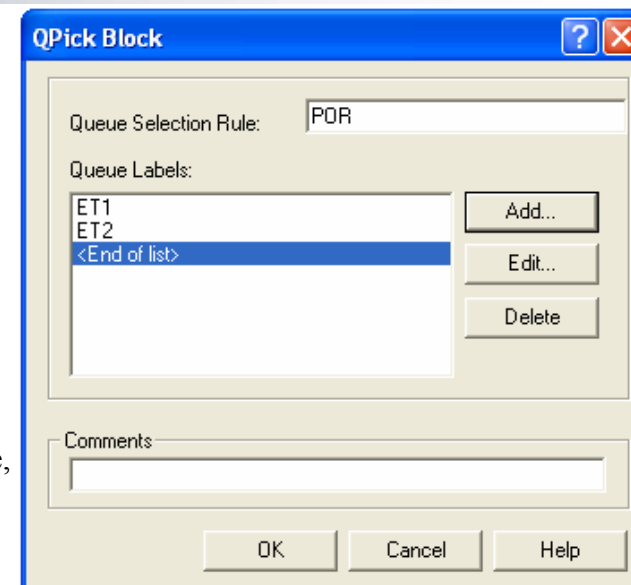
POR - conform acestei reguli ARENA alege primul șir disponibil (ordinea șirurilor este stabilită de ordinea în care apar etichetele acestora în lista din caseta *Queue Labels*),

RAN - șirul este ales în mod aleator,

SNQ - entitatea activă va fi trimisă spre șirul cu cele mai puține poziții ocupate,

SRC - șirul ales este cel cu numărul cel mai mic de poziții libere,

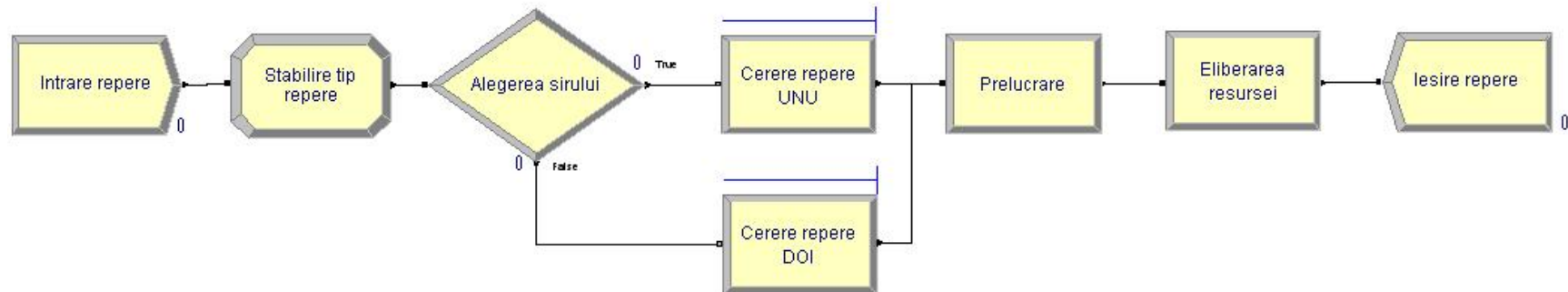
ER sau UR - utilizatorul specifică regula de selecție cu ajutorul elementului *RULE* sau prin intermediul unei funcții.



Exemplu de utilizare a blocului QPick

- În cadrul unei stații de prelucrare, formată dintr-o singură mașină, sunt prelucrate două tipuri de repere. Piese aferente acestor tipuri intră în stația de prelucrare, una câte una, la intervale de timp ale căror valori sunt distribuite după o lege exponențială cu media de 5 minute. După intrarea în sistem, reperele se atașează dacă este cazul unuia dintre cele două șiruri aflate în fața mașinii, fiecare tip de reper având asociat un șir de așteptare propriu. Pe baza datelor culese în timp s-a observat că reperele prelucrate în sistemul de față aparțin în proporție de 80% primului tip de reper. În plus, reperele din această primă categorie sunt prelucrate cu prioritate față de cele care aparțin celei de-a doua. Acest fapt înseamnă că prelucrarea reperelor de tipul al doilea nu începe dacă în sistem există repere aparținând primului tip. Totuși prelucrarea unui reper nu este întreruptă de intrarea în sistem a unui reper cu prioritate mai ridicată. Durata prelucrării are valori distribuite după o lege normală cu media de 4 minute și abaterea medie pătratică de 2 minute.
- Simulați activitatea sistemului pe durata a 480 minute.

Exemplu de utilizare a blocului QPick



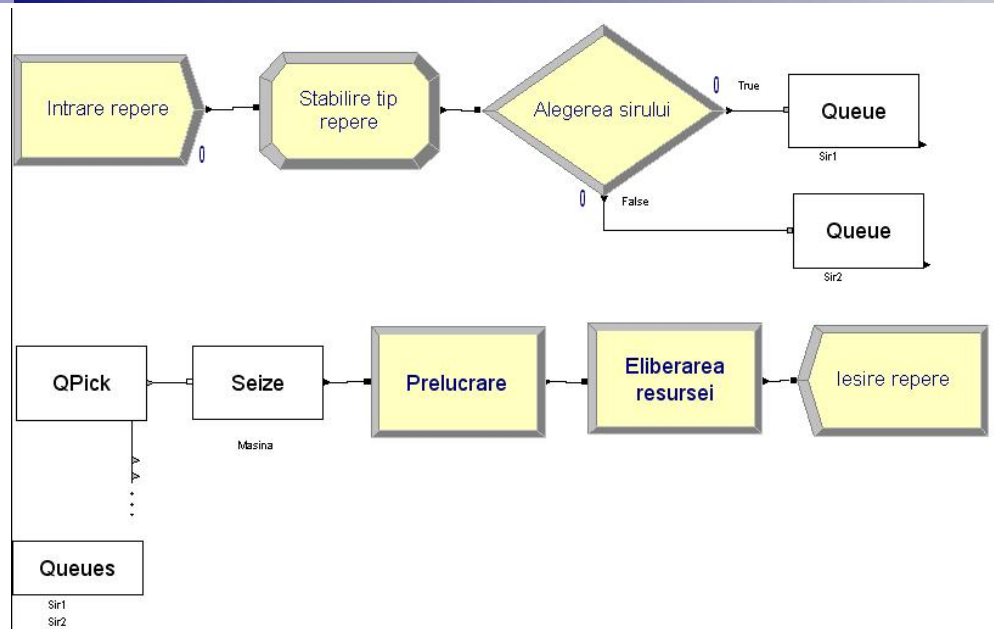
```

; Model statements for module: Create 1
9$ CREATE, 1,MinutesToBaseTime(0.0),Entity
  1:MinutesToBaseTime(EXPO(5)): NEXT(10$);
10$ ASSIGN: Intrare repere.NumberOut=Intrare repere.NumberOut +
  1:NEXT(0$);
; Model statements for module: Assign 1
0$ ASSIGN: Tip_repere=DISCRETE(0.8,1,1.0,2):
  Entity.Picture=(Tip_repere==1)*5+(Tip_repere==2)*8: NEXT(1$);
; Model statements for module: Decide 2
1$ BRANCH, 1:
  If,Tip_repere==1,13$,Yes:
  Else,14$,Yes;
13$ ASSIGN: Alegerea sirului.NumberOut True=Alegerea
  sirului.NumberOut True + 1:NEXT(5$);
14$ ASSIGN: Alegerea sirului.NumberOut False=Alegerea
  sirului.NumberOut False + 1:NEXT(7$);
; Model statements for module: Seize 1
5$ QUEUE, Sir1;
  SEIZE, 1,Other: Masina,1:NEXT(16$);
16$ DELAY: 0.0,,VA:NEXT(2$);
  
```

```

; Model statements for module: Delay 1
2$ DELAY: MinutesToBaseTime(NORM( 4, 2)),,Other:NEXT(3$);
; Model statements for module: Release 1
3$ RELEASE: Masina,1:NEXT(4$);
; Model statements for module: Dispose 1
4$ ASSIGN: Iesire repere.NumberOut=Iesire
  repere.NumberOut + 1;
17$ DISPOSE: Yes;
; Model statements for module: Seize 2
7$ QUEUE, Sir2;
  SEIZE, 2,Other: Masina,1:NEXT(19$);
19$ DELAY: 0.0,,VA:NEXT(2$);
  
```

Exemplu de utilizare a blocului QPick



```

; Model statements for module: Create 1
6$ CREATE, 1,MinutesToBaseTime(0.0),Entity
1:MinutesToBaseTime(EXPO(5)):NEXT(7$);
7$ ASSIGN: Intrare repere.NumberOut=Intrare
repere.NumberOut + 1:NEXT(0$);
; Model statements for module: Assign 1
0$ ASSIGN: Tip_repere=DISCRETE(0.8,1,1.0,2):
Entity.Picture=(Tip_repere==1)*5+(Tip_repere==2)*8:
NEXT(1$);
    
```

```

; Model statements for module: Decide 2
1$ BRANCH, 1:
If,Tip_repere==1,10$,Yes:
Else,11$,Yes;
10$ ASSIGN: Alegerea sirului.NumberOut True=Alegerea
sirului.NumberOut True + 1:NEXT(Eticheta_1);
11$ ASSIGN: Alegerea sirului.NumberOut False=Alegerea
sirului.NumberOut False + 1:NEXT(Eticheta_2);
    
```

```

Eticheta_1 QUEUE, Sir1:DETACH;
Eticheta_2 QUEUE, Sir2:DETACH;
    
```

```

QPICK, POR:
Eticheta_1:
Eticheta_2;
    
```

```

5$ SEIZE, 1,Other: Masina,1:NEXT(2$);
; Model statements for module: Delay 1
2$ DELAY: MinutesToBaseTime(NORM( 4,
2)),Other:NEXT(3$);
; Model statements for module: Release 1
3$ RELEASE: Masina,1:NEXT(4$);
; Model statements for module: Dispose 1
4$ ASSIGN: Iesire repere.NumberOut=Iesire
repere.NumberOut + 1;
12$ DISPOSE: Yes;
    
```