

Laboratory 7 MS

Problem 1:

1. Fie un sistem de operare de timp real care trebuie să planifice la execuție două tipuri de taskuri ce vor fi executate la intervale egale. Taskurile de primul tip sunt generate la intervale de 2s, cele de tipul doi la intervale de 5s. Taskurile vor fi generate de două blocuri Entity Generation. Durata de deservire a taskurilor de primul tip este de 1s, a celor de tipul doi este 2.5s. Se va simula modelul în următoarele cazuri:

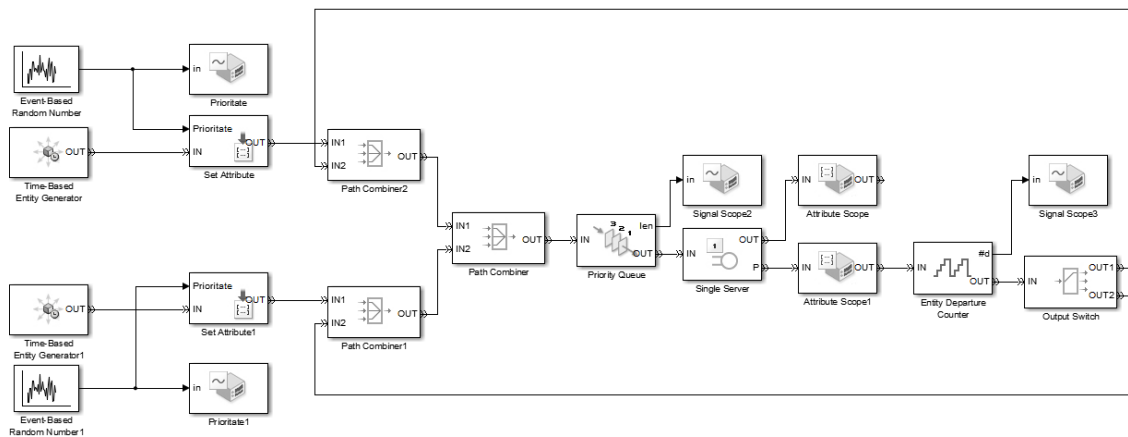
- Ambele tipuri de taskuri au aceeași prioritate
- Taskurile de primul tip au prioritate mai mare
- Taskurile de tipul doi au prioritate mai mare.

Modelul se va simula pe o durată de 50s. Unitatea de timp este secunda. Se va considera cazul când entitatea preluată reîntră în server cu timpul de deservire rezidual.

Entitățile vor avea următoarele atribute: prioritatea, timpul de deservire și tipul taskului.

Se vor înregistra: prioritatea entităților generate, lungimea cozii, timpul de deservire rezidual, numărul de entități preluate și numărul de entități deservite pentru fiecare tip de task. Se vor explica rezultatele.

We designed the following schema:



First entity generator that produces entities of type 1 every two seconds:

Block Parameters: Time-Based Entity Generator

Time-Based Entity Generator

Generate entities using intergeneration times from a signal or a statistical distribution.

Entity Generation Blocking Entity Type Statistics

Generate entities upon: Intergeneration time from dialog

Distribution: Constant

Period: 2

Generation event priority: 300

☒ Generate entity at simulation start

Second entity generator (5 seconds):

Block Parameters: Time-Based Entity Generator1

Time-Based Entity Generator

Generate entities using intergeneration times from a signal or a statistical distribution.

Entity Generation Blocking Entity Type Statistics

Generate entities upon: Intergeneration time from dialog

Distribution: Constant

Period: 5

Generation event priority: 300

☒ Generate entity at simulation start

The priority is decided in a random number generator

Source Block Parameters: Event-Based Random Number

Event-Based Random Number

Generate random numbers from the specified distribution, parameters, and initial seed.

Parameters

Distribution: Exponential

Mean: 1

Initial seed: 12344

OK Cancel Help Apply

Source Block Parameters: Event-Based Random Number1

Event-Based Random Number

Generate random numbers from the specified distribution, parameters, and initial seed.

Parameters

Distribution: Exponential

Mean: 1

Initial seed: 12448

OK Cancel Help Apply

Serving time for the first task:

2	Timp_deservire	Dialog	1	<input checked="" type="checkbox"/>
---	----------------	--------	---	-------------------------------------

Serving time for the second task:

The server settings:

Block Parameters: Single Server

Single Server

Serve one entity for a period of time and then attempt to output the entity through the OUT port. If the OUT port is blocked then the entity stays in this block until the port becomes unblocked. You can specify the service time, which is the duration of service, via a parameter, attribute, or signal.

When the block permits preemption, an entity in the server can depart early via the P port. Preemption occurs only if attributes of the current entity and the entity attempting to arrive satisfy specified criteria.

When the block does not permit preemption, the IN port is unavailable whenever this block stores an entity.

Single Server

Preemption

Timeout

Statistics

Service time from: Attribute

Attribute name: Timp_deservire

Service completion event priority: 500

The priority queue:

Block Parameters: Priority Queue

Priority Queue

Store entities in sorted sequence for an undetermined length of time. The Capacity parameter is the number of entities the queue can hold. The queue sorts entities according to the values of the specified attribute, in either ascending or descending order.

Priority Queue

Timeout


Statistics

Capacity: 25

Sorting attribute name: Prioritate

Sorting direction: Ascending

The entity departure counter:

 Block Parameters: Entity Departure Counter ✕

Entity Departure Counter
Count departures and write the total to a signal port and/or to an attribute of the departing entity.


Parameters
☒ Write count to signal port #d
☒ Write count to attribute
Attribute name:

Count ▾

☒ Create attribute if not present
Reset counter upon:

Off ▾

The output switch depending on the task:

 Block Parameters: Output Switch ✕

Output Switch
Select an entity output port for departure. The Switching criterion parameter indicates how the block determines which entity output port is selected for departure at any given time.

The block receives entities and outputs them through one of the entity output ports. The port selected for departures can change during the simulation. When the selected entity output port is not blocked, an arriving entity departs through this port.

Output Switch

Timeout

Statistics

Number of entity output ports:

2

Switching criterion:

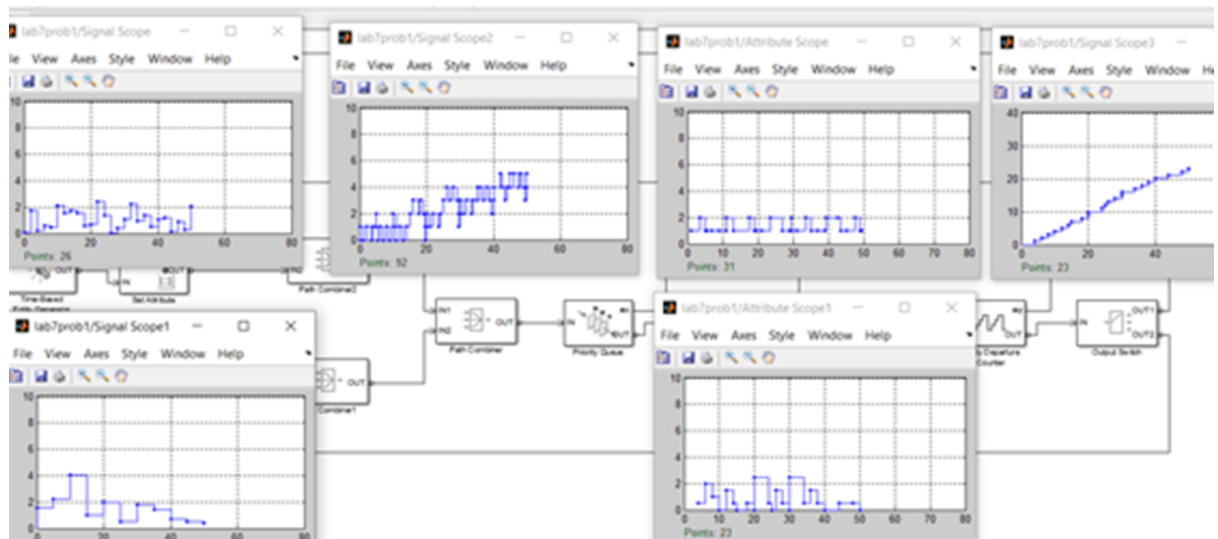
From attribute ▾

Attribute name:

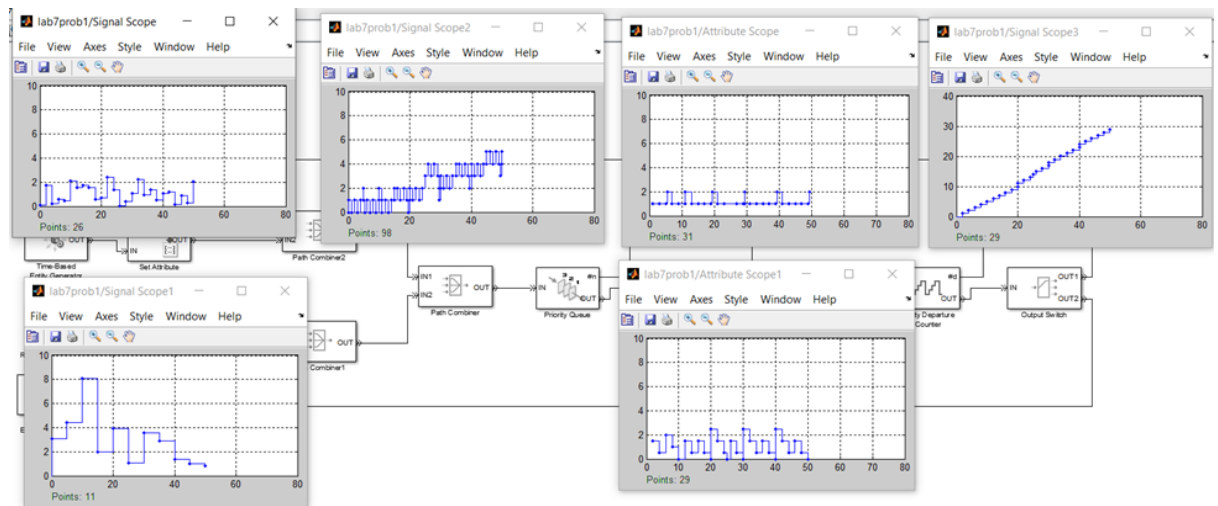
Tip_task ▾

We got the following results:

Equal priority



First task with higher priority:



Second task with higher priority:

