

Извјештај за трећу лабораторијску вјежбу

из

Програмирања у реалном времену

1. задатак

а) Нека је дат скуп периодичних задатака $\Gamma = \{a, b, c\}$ који се извршавају на једном процесору, при чему су вриједности периода задатака, њихова времена извршавања и крајњи рокови дати у Табели 1.1 тако да колона са ознаком T_i садржи периоде задатка, колона C_i садржи времена извршавања задатка, а колона D_i крајње рокове до којих се задатак мора извршити.

Табела 1.1

бр.	задатак	T_i [ms]	C_i [ms]	D_i [ms]
1	a	10	5	10
2	b	20	20	20
3	c	30	15	30

Овако задат скуп задатака Γ није распоредив ни по *EDF* (енг. *Earliest Deadline First*) ни по *RM* (енг. *Rate Monotonic*) алгоритму за распоређивање периодичних задатака.

Ако израчунамо заузеће процесора, U , за скуп задатака Γ , добијамо следеће резултате:

$$U = \sum_{i=1}^3 \frac{C_i}{T_i} = \frac{5}{10} + \frac{20}{20} + \frac{15}{30} = 2$$

Како је $U > 1$, то значи да овај скуп задатака није распоредив.

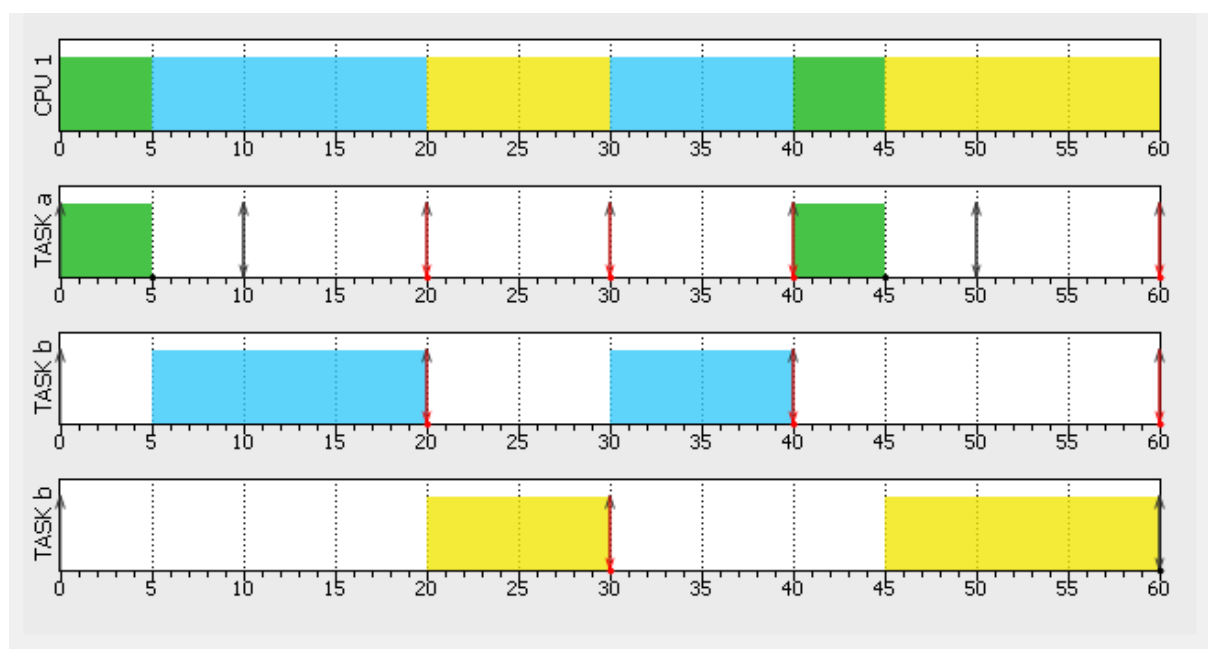
Овре резултате потврђује и симулација у *SimSo* програму. Симулација траје у дужини једног хиперпериода и он за дати скуп задатака Γ износи:

$$H = NZS(T_1, T_2, T_3)$$

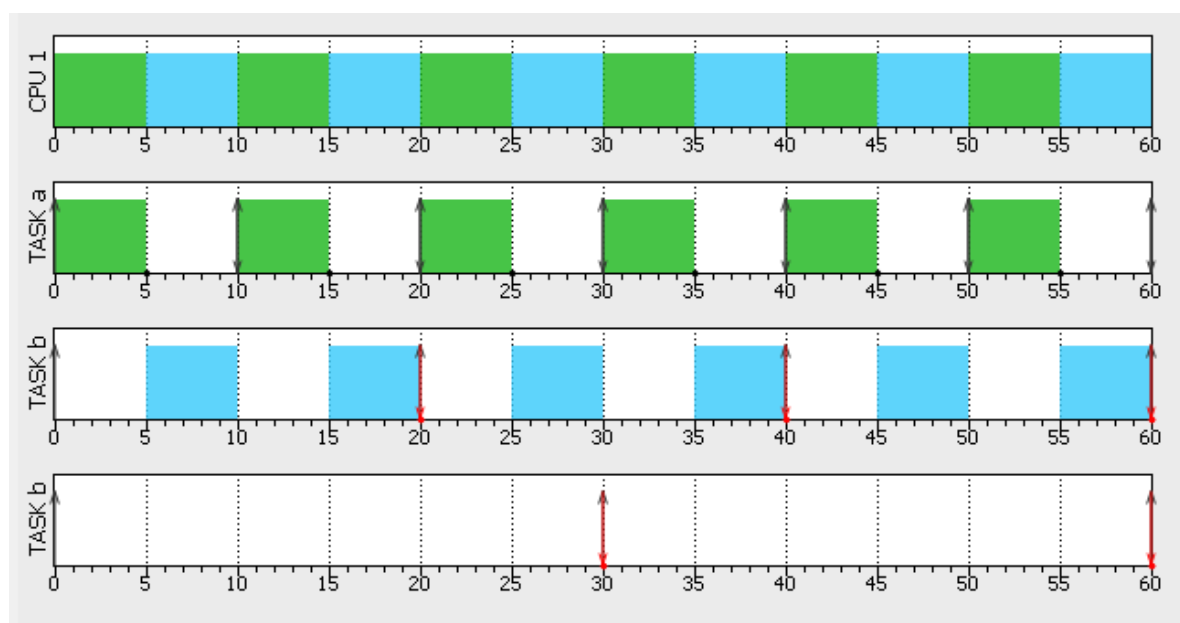
$$H = 60$$

,при чему *NZS* означава најмањи заједнички садржалац.

Гантов дијаграм добијен симулацијом у случају распоређивања *EDF* алгоритмом дат је на Слици 1.1 док је Гантов дијаграм у случају распоређивања *RM* алгоритмом дат на Слици 1.3. Црвеним линијама на Гантовим дијаграмима означени су тренуци у симулацији у којима су нарушени крајњи рокови за извршавање задатака. Примјећујемо да је за вријеме трајања читавог хиперпериода процесор заузет извршавањем задатака, што је у складу са теоретским очекивањима која смо претходно добили.



Слика 1.1



Слика 2.1

б) Нека је дат скуп периодичних задатака $\Gamma = \{a, b, c\}$ који се извршавају на једном процесору, при чему су вриједности периода задатака, њихова времена извршавања и крајњи рокови дати у Табели 1.2.

Табела 2.2

бр.	задаатак	T_i [ms]	C_i [ms]	D_i [ms]
1	a	5	2	5
2	b	10	3	10
3	c	15	4.5	15

Ако израчунамо заузеће процесора, U , за скуп задатака Γ , добијамо следеће резултате:

$$U = \sum_{i=1}^3 \frac{C_i}{T_i} = \frac{2}{5} + \frac{3}{10} + \frac{5}{15} = 1$$

Како је $U = 1$, скуп задатака Γ је распоредив по EDF алгоритму, јер је потребан и довољан услов распоредивости за овај алгоритам $U \leq 1$ што је испуњено. Псматрајмо времена одзива задатака из скупа Γ :

Вријеме одзива задатка a :

$$R_a = 2 < T_1$$

Вријеме одзива задатка b :

$$w_b^0 = 3$$

$$w_b^1 = 3 + \left\lceil \frac{3}{5} \right\rceil \cdot 2 = 3 + 2 = 5$$

$$w_b^2 = 3 + \left\lceil \frac{5}{5} \right\rceil \cdot 2 = 5$$

$$R_b = 5 < T_2$$

Вријеме одзива задатка c :

$$w_c^0 = 4.5$$

$$w_c^1 = 4.5 + \left\lceil \frac{4.5}{5} \right\rceil \cdot 2 + \left\lceil \frac{4.5}{10} \right\rceil \cdot 3 = 11.85$$

$$w_c^2 = 4.5 + \left\lceil \frac{11.85}{5} \right\rceil \cdot 2 + \left\lceil \frac{11.85}{10} \right\rceil \cdot 3 = 22.5$$

$$w_c^3 = 4.5 + \left\lceil \frac{22.5}{5} \right\rceil \cdot 2 + \left\lceil \frac{22.5}{10} \right\rceil \cdot 3 = 23.5$$

$$w_c^4 = 4.5 + \left\lceil \frac{23.5}{5} \right\rceil \cdot 2 + \left\lceil \frac{23.5}{10} \right\rceil \cdot 3 = 23.5$$

$$R_c = 23.5 > T_3$$

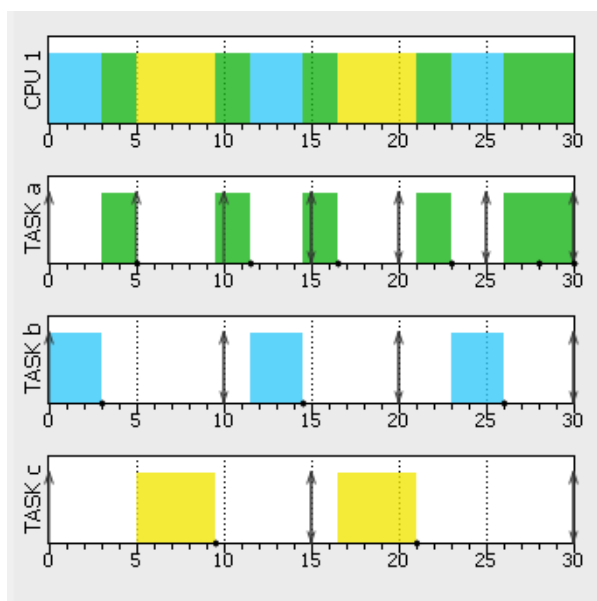
Како је најгоре вријеме одзива за задатак c веће од његовог периода слиједи да скуп задатака Γ није распоредив по RM алгоритму.

Хиперпериод за скуп задатака Γ је:

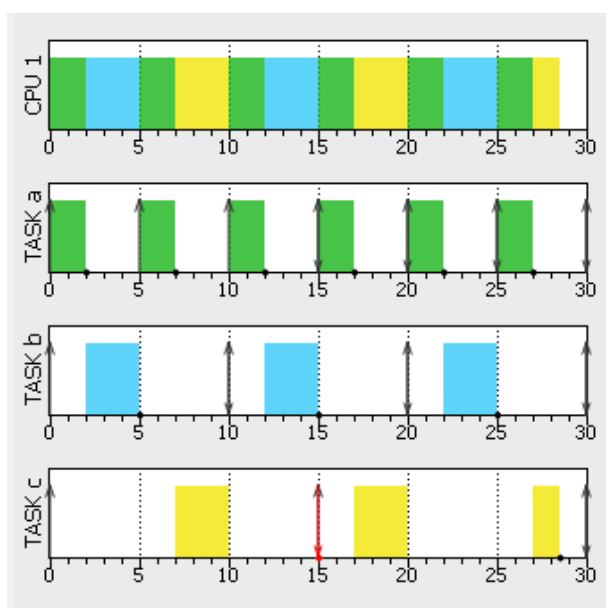
$$H = NZS(T_1, T_2, T_3)$$

$$H = 30$$

Резултати симулације добијени помоћу програма *SimSo* у трајању једног хиперпериода, потврђују добијене теоретске резултате. На Слици 1.2 и Слици 1.3 дати су Гантови дијаграми у случају да се задаци из скупа Γ распоређују EDF и RM , респективно.



Слика 1.3



Слика 1.4

Примјећујемо да је црвеном линијом у случају распоређивања *RM* алгоритмом означено мјесто на Гантовом дијаграму када долази до нарушавања крајњег рока извршавања за задатак *c*. На Слици 1.5 и Слици 1.6 дати су резултати симулације који показују заузеће процесора за *EDF* и *RM*, респективно.

Observation Window:			
from 0.00 to 30.00 ms		Configure...	
	Total load	Payload	System load
CPU 1	1.0000	1.0000	0.0000
Average	1.0000	1.0000	0.0000

Слика 1.5

General	Logs	Tasks	Scheduler	Processors
Observation Window:				
from 0.00 to 30.00 ms		Configure...		
	Total load	Payload	System load	
CPU 1	0.9500	0.9500	0.0000	
Average	0.9500	0.9500	0.0000	

Слика 1.6

ц) Нека је дат скуп периодичних задатака $\Gamma = \{a, b, c\}$ који се извршавају на једном процесору, при чему су вриједности периода задатака, њихова времена извршавања и крајњи рокови дати у Табели 1.3.

Табела 3.3

бр.	задаатак	T_i [ms]	C_i [ms]	D_i [ms]
1	a	10	5	10
2	b	20	5	20
3	c	30	2	30

Овакав скуп задатака распоредив је и са EDF и са RM алгоритмом. Што показују теоретска очекивања и резултати симулације у програму *SimSo*.

Ако израчунамо заузеће процесора, U , за скуп задатака Γ , добијамо следеће резултате:

$$U = \sum_{i=1}^3 \frac{C_i}{T_i} = \frac{5}{10} + \frac{5}{20} + \frac{2}{30} \approx 0.8167$$

Како је $U \leq 1$ имамо да је овај скуп задатака EDF распоредив. Погледајмо времена одзива за задатке из скупа Γ :

Вријеме одзива задатка a :

$$R_a = 5 < T_1$$

Вријеме одзива задатка b :

$$w_b^0 = 5$$

$$w_b^1 = 5 + \left\lceil \frac{5}{10} \right\rceil \cdot 5 = 5 + 5 = 10$$

$$w_b^2 = 5 + \left\lceil \frac{10}{10} \right\rceil \cdot 5 = 5 + 5 = 10$$

$$R_b = 10 \leq T_2$$

Вријеме одзива задатка c :

$$w_c^0 = 2$$

$$w_c^1 = 2 + \left\lceil \frac{2}{10} \right\rceil \cdot 5 + \left\lceil \frac{2}{20} \right\rceil \cdot 5 = 2 + 5 + 5 = 12$$

$$w_c^2 = 2 + \left\lceil \frac{12}{10} \right\rceil \cdot 5 + \left\lceil \frac{12}{20} \right\rceil \cdot 5 = 2 + 10 + 5 = 17$$

$$w_c^3 = 2 + \left\lceil \frac{17}{10} \right\rceil \cdot 5 + \left\lceil \frac{17}{20} \right\rceil \cdot 5 = 17$$

$$R_c = 17 < T_3$$

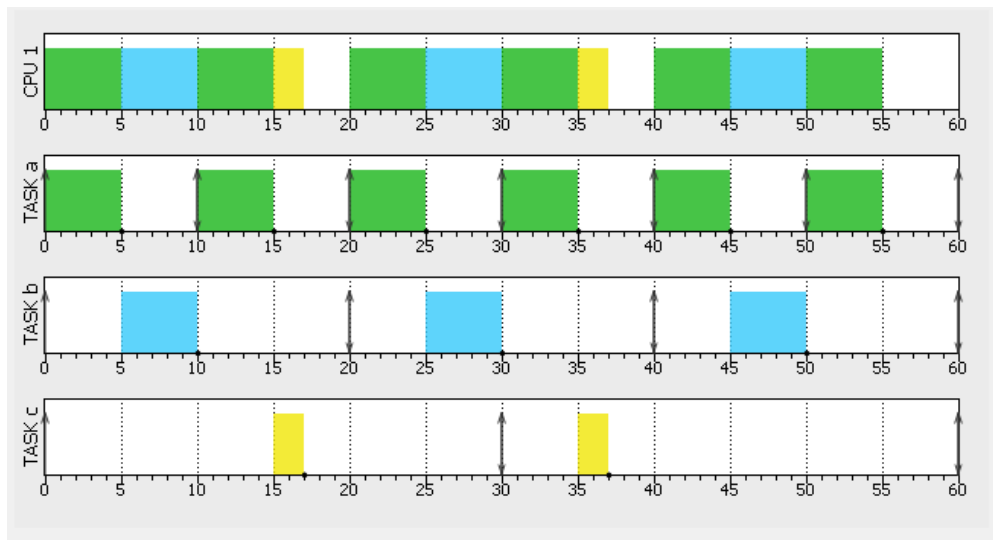
Будући да су за све задатке времена одзива мања од њихових респективних периода имамо да је скуп задатака Γ RM распоредив.

Хиперпериод за скуп задатака Γ је:

$$H = NZS(T_1, T_2, T_3)$$

$$H = 60$$

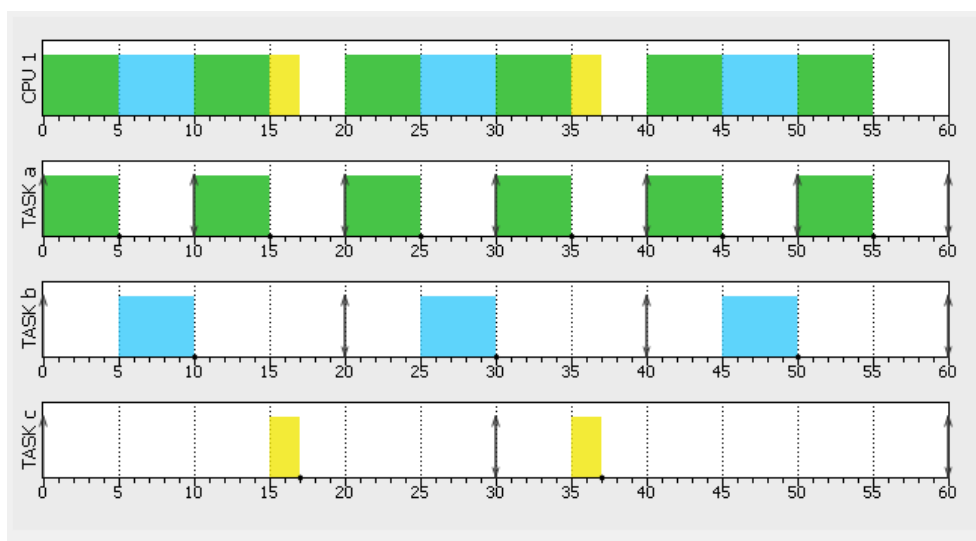
Заузеће процесора и Гантов дијаграм за распоређивање EDF и RM алгоритмом дати су на Слици 1.7, Слици 1.8 и Слици 1.9, Слици 2.0, респективно.



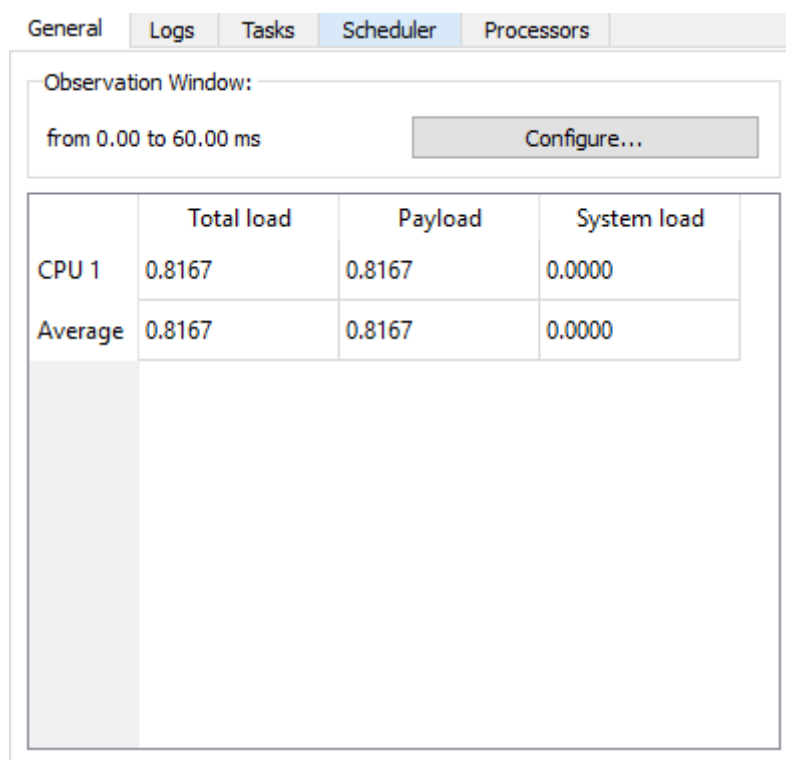
Слика 1.7

General			
Logs			
Tasks			
Scheduler			
Processors			
Observation Window:			
from 0.00 to 60.00 ms			Configure...
	Total load	Payload	System load
CPU 1	0.8167	0.8167	0.0000
Average	0.8167	0.8167	0.0000

Слика 1.8



Слика 1.9



Слика 2.0








Примјећујемо да су симулације успјешно извршене у трајању од једног хиперперода, те дасу добијени резултати у складу са теоретским очекивањима.

2. задатак

Нека је дат скуп задатака $\Gamma = \{a, b, c, d, e\}$ који се извршавају на једном процесору, при чему су вриједности периода задатака, њихова времена извршавања и крајњи рокови дати у Табели 1.3. Задаци d и e су апериодични. Задаци се распоређују RM алгоритмом при, а симулација распоређивања је извршена у *Timestool* програму. На Слици 2.1 приказана је табела са задацима из *Timestool*-а при чему задацима $task1, task2, task3, task4, task5$ одговарају задаци a, b, c, d, e из скупа Γ , респективно.

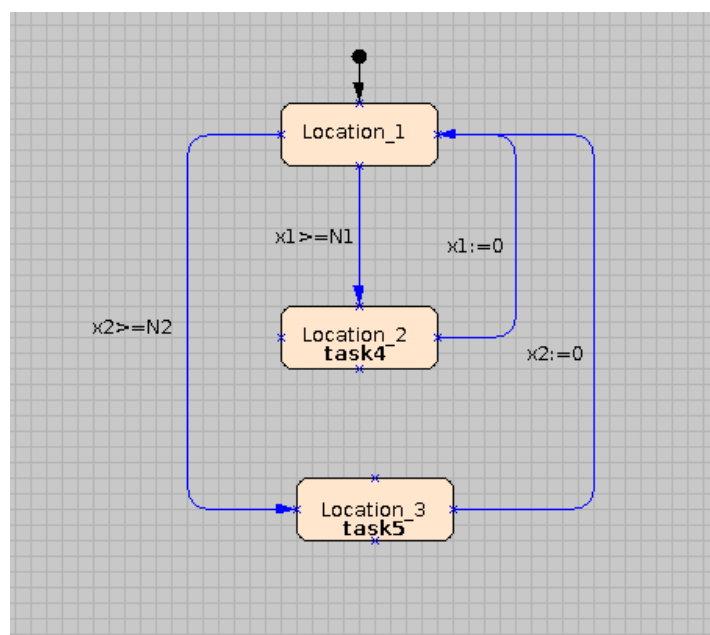
Табела 4.3

бр.	задатак	T_i [ms]	C_i [ms]	D_i [ms]
1	a	10	5	10
2	b	20	5	20
3	c	30	2	30
4	d	-	1	5
5	e	-	1	5

Name	B	Pr	C	D	T
 task1	P	3	5	10	10
 task2	P	2	5	20	20
 task3	P	1	2	30	30
 task4	C	5	1	5	
 task5	C	4	1	5	

Слика 2.1

За распоређивање апериодичних задатака користи се машина стања приказана на Слици 2.3 чији су параметри дати на Слици 2.4.



Слика 2.3

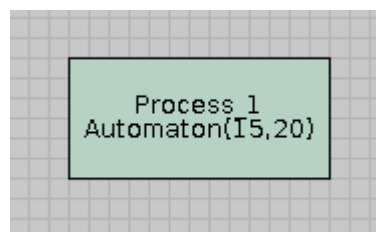
Properties		
Template attributes		
Name	Automaton	
Parameters	const N1,const N2	
<input type="checkbox"/> Environment		
Local declarations		
Name	Type	Value
x1	clock	0
x2	clock	0

Слика 2.4

Чвор *Location_1* представља иницијално стање у којем се налази машина стања *Automaton*. Када промјенљива *x1* достигне вриједност *N1* машина стања прелази из иницијалног у стање *Location_2* гдје се врши распоређивање задатка *d* и потом се долази до повратка у иницијално стање и ресетовања вриједности *x1* на нулу.

Када промјенљива *x2* достигне вриједност *N2* машина стања прелази из иницијалног у стање *Location_3* гдје се врши распоређивање задатка *e* и потом се долази до повратка у иницијално стање и ресетовања вриједности *x2* на нулу.

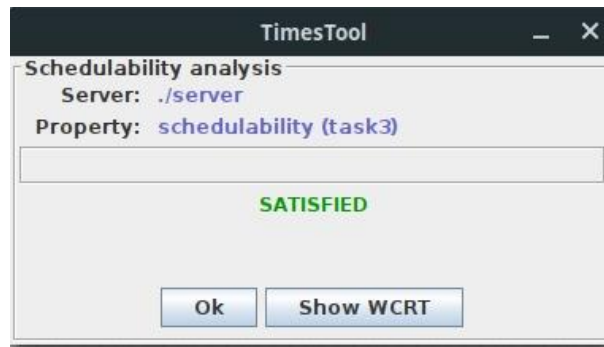
Машина стања је представљена процесом *Process_1* у којем се задају вриједности за константе *N1* и *N2*. У овом случају ради распоредивости задатака одабране су вриједности *N1*=15 и *N2*=20, што је приказано на Слици 2.5.



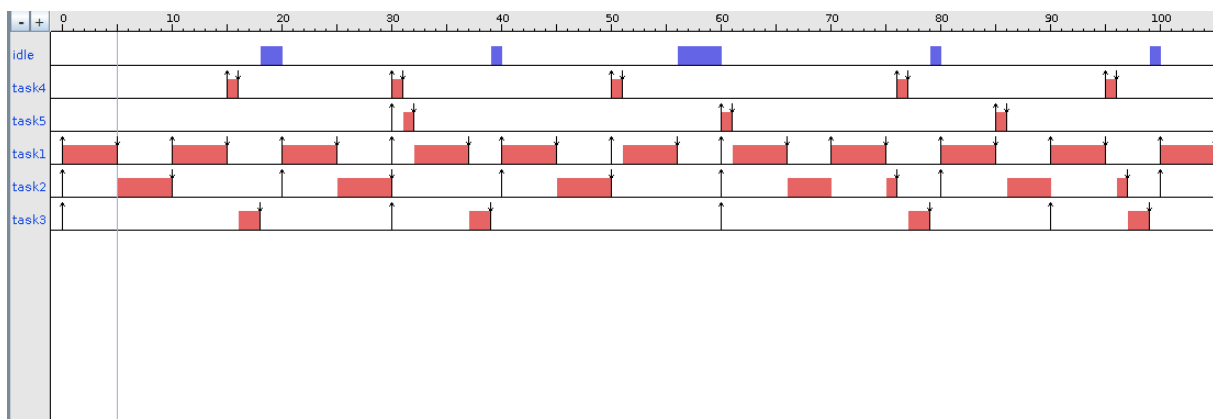
Слика 2.5

Задаци са овако подешеним параметрима су распоредиви што потврђује верификација распоредивост *Timestool* алата са Слике 2.6, а приказ симулације њиховог извршавања дат је на Слици 2.7. Из симулације се види да је заузеће процесора сигурно мање од

100%, будући да се на Гантовом дијаграму уочавају *idle* дијелови у којима је процесор слободан и не извршава ни један задатак.



Слика 2.6



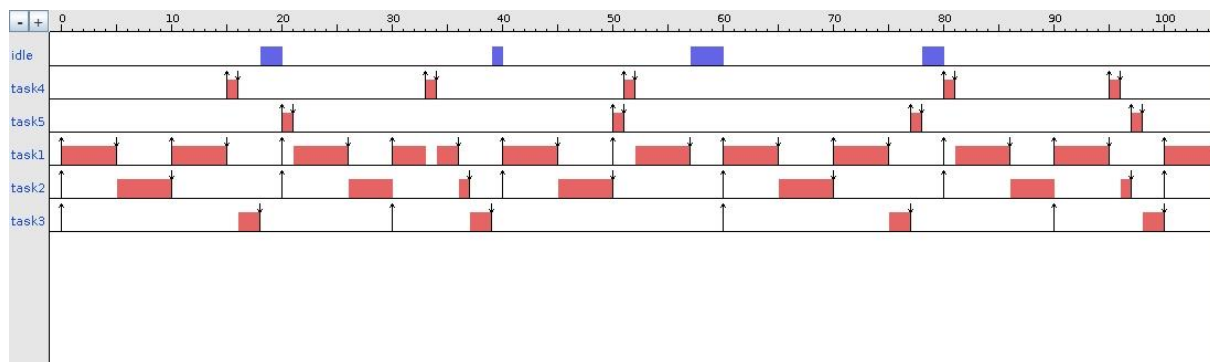
Слика 2.7

У наставку разматрамо случај када постоји зависност између задатака. Према поставци задатка додата је зависност између два произвољна задатка, те према томе задатак *b* зависи од задатка *a* што је приказано на Слици 2.8.



Слика 2.8

У овом конкретном случају верификација распоредивости не даје коначне резултате већ се провјера не завршава у неком разумном коначном времену, међутим при покретању симулације у трајању већем од 5 *min* није дошло до било каквог незадовољавања рокова. На Слици 2.9 приказан је резултат симулације.



Слика 2.9