

Universidad Nacional
ARTURO JAURETCHE

PROGRAMACIÓN EN TIEMPO REAL

TRABAJO PRÁCTICO N° 2

Planificación de procesos

Profesor

Diego ENCINAS

Autores

Cristian SANABRIA

Gerónimo BAZÁN

Emiliano SALVATORI

17 de octubre de 2020

Índice

1. Introducción	2
2. Cuestionario	2
2.1. Ejercicio nº 1	2
2.2. Ejercicio nº 2	2
2.3. Ejercicio nº 3	2
3. Resoluciones	3
3.1. Ejercicio nº 1	3
3.2. Ejercicio nº 2	5
3.3. Ejercicio nº 3	6

1. Introducción

En el siguiente informe se detalla lo realizado como parte del laboratorio de la materia **Programación en Tiempo Real** para la **Comisión nº 1**. El presente trabajo se basa en lo solicitado para la Práctica nº 2 sobre **Planificación de procesos** realizando una investigación y simulación con dos tipos de planificaciones.

2. Cuestionario

2.1. Ejercicio nº 1

En un sistema de tiempo real con las siguientes tareas:

Tarea	Tiempo de ejecución	Período/Plazo
A	2	12
B	2	24
C	2	6
D	1	3

1. Compruebe si existe una planificación de tiempo real viable.
2. Desarrolle un plan cíclico para la ejecución de estas tareas, calculando el período principal y secundario como también mostrando la tabla de tareas para el ejecutivo cíclico.

2.2. Ejercicio nº 2

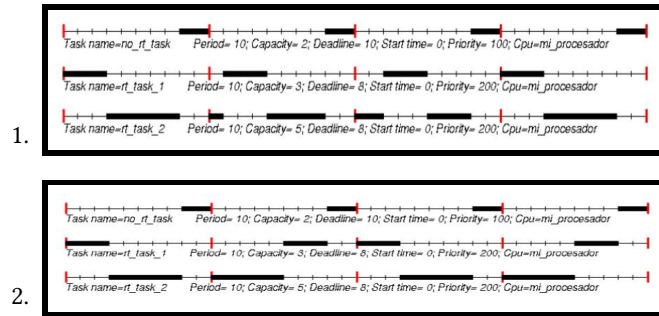
En un sistema de tiempo real con las siguientes tareas:

Tarea	Tiempo de ejecución	Período/Plazo
A	1	10
B	3	12
C	7	20
D	1	5

1. Compruebe si existe una planificación de tiempo real viable.
2. Desarrolle un plan cíclico para la ejecución de estas tareas, calculando el período principal y secundario como también mostrando la tabla de tareas para el ejecutivo cíclico.

2.3. Ejercicio nº 3

Encuentre la parametrización correcta para que la simulación de **Cheddar**, usando el algoritmo de planificación *POSIX 1003.1b/Highest Priority First* resulte de la siguiente forma:



3. Resoluciones

3.1. Ejercicio nº 1

A continuación se desarrolla la solución del **Ejercicio nº 1**:

1. Para comprobar si es posible una planificación de Tiempo Real viable, se tener en cuenta la siguiente ecuación:

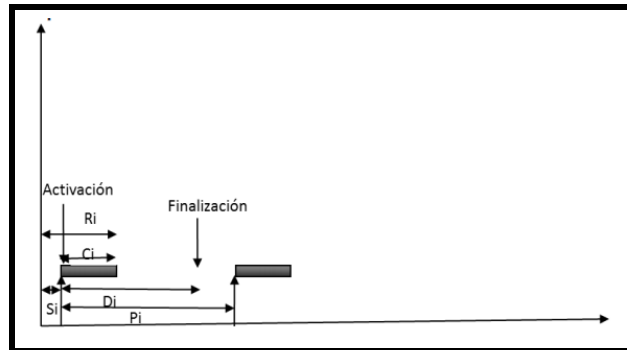
$$U = \sum_{i=1}^N \frac{C_i}{P_i}$$

:

El resultado de esta ecuación debe ser *menor o igual a uno* para saber si la planificación es viable. Se deben tener en cuenta los siguientes parámetros:

- **N**: Número de Tareas en el sistema.
- **P_i**: Período de activación.
- **C_i**: Tiempo máximo de ejecución.
- **D_i**: Plazo máximo de terminación.
- **R_i**: Tiempo de respuesta máximo. Es el tiempo desde que la tarea se activa hasta que termina realmente de ejecutarse.
- **Pri**: Prioridad de la tarea (convenio: mayor valor, menor prioridad).
- **Si**: Desfase respecto al momento inicial.

Se puede tener en cuenta la siguiente imagen para comprender en profundidad lo anterior:



Una vez que se tiene en cuenta lo anterior, se procede a resolver la ecuación:

$$U = \frac{2}{12} + \frac{2}{24} + \frac{2}{6} + \frac{1}{3}$$

Calculamos el común múltiplo de la ecuación:

$$U = \frac{4}{24} + \frac{2}{24} + \frac{8}{24} + \frac{8}{24}$$

Obteniendo:

$$U = 0,916$$

Por lo cual, se llega a la conclusión que la planificación es viable.

2. El ciclo principal también conocido como *Hiperperíodo* se calcula con la siguiente fórmula:

$$T_m = mcm(T_i)$$

Donde se entiende que T_m es el igual al mínimo común múltiplo de los períodos de todas las tareas, es decir que el valor es **24**.

El ciclo secundario se puede comprender mediante la siguiente igualdad:

$$T_s | T_m = k * T_s$$

Donde el Ciclo Secundario (T_s) es un factor (o simplemente *divide a*) del Hiperperíodo (T_m), donde este último se puede ver como una constante (k) que multiplica al Ciclo Secundario. Hay que tener en cuenta una de las propiedades de los ejecutivos cíclicos, la cual establece que *todos los períodos de los procesos deben ser múltiplos del tiempo del ciclo secundario*. Teniendo esto en cuenta, se obtiene:

$$T_s | T_m = k * T_s$$

$$T_s | 24 = 8 * 3$$

$$T_s = 3$$

Una vez calculados los periodos, se procede a realizar la tabla de tareas correspondiente a cada una de ellas:

Tarea	Tiempo de ejecucion	Periodo/Plazo
A	2	12
B	2	24
C	2	6
D	1	3

Y luego se procede a insertar cada uno de los procesos dividiéndolo según el esquema obtenido:

Ejecutivo Cíclico																							
T _m = 24																							
T _s = 3			T _s = 3			T _s = 3			T _s = 3			T _s = 3			T _s = 3			T _s = 3			T _s = 3		
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
D	C	C	D	A	A	D	C	C	D	B	B	D	C	C	D	A	A	D	C	C	D		

Cabe aclarar que las veces que las tareas aparecen en el gráfico son de dividir el *Hiperperiodo* (T_m) por el periodo de cada tarea.

3.2. Ejercicio nº 2

1. Nuevamente, para comprobar si es posible una planificación de Tiempo Real viable, se debe tener en cuenta la ecuación vista en la sección 1:

$$U = \sum_{i=1}^N \frac{C_i}{P_i}$$

Procedemos a resolver según los valores obtenidos de la tabla:

$$U = \frac{1}{10} + \frac{3}{12} + \frac{7}{20} + \frac{1}{5}$$

Nuevamente calculamos el mínimo común múltiplo y resolvemos:

$$U = \frac{6}{60} + \frac{15}{60} + \frac{21}{60} + \frac{12}{60}$$

Obteniendo como resultado:

$$U = 0,9$$

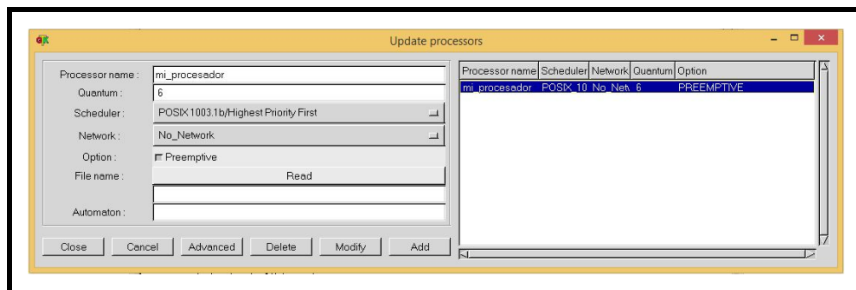
Llegando también a que la planificación es viable.

2. En este ejercicio se considera que el *Hiperperíodo* (T_m) es 60 pero el Ciclo Secundario (T_s) no se puede obtener, ya que no respeta una de las propiedades de los ejecutivos cíclicos que dictamina: *todos los periodos de los procesos deben ser múltiplos del tiempo del Ciclo Secundario*, es decir, ningún numero multiplicado por una constante k puede dar como resultado ser múltiplo de todos los periodos de los procesos. Por lo cual, no se puede realizar el grafico del ejecutivo cíclico.

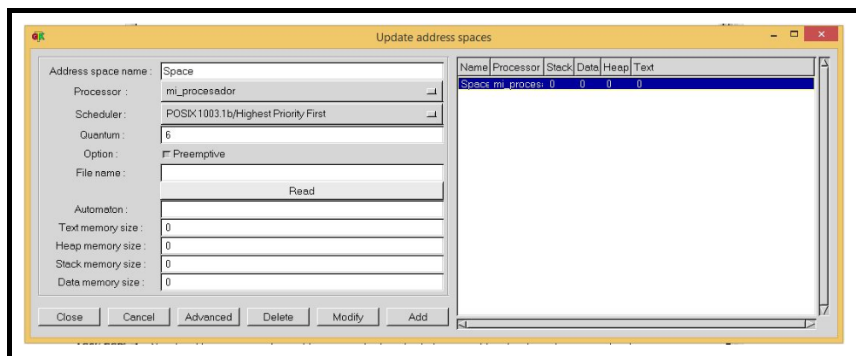
3.3. Ejercicio nº 3

Cabe mencionar que para realizar la simulación, se debe operar en la versión de *Cheddar*, 2.1.

1. Para realizar este inciso, lo primero que se debe llevar a cabo es la configuración de un procesador; para ello se debe ir a la solapa *edit* y luego a *Update Processors*. Una vez dentro se debe configurar el Quantum a 6 ya que las tareas deben ejecutarse con la política de Round Robin.



Una vez configurado el procesador, se debe configurar el *Address Space*, como se puede ver a continuación:



Luego de configurar el *Address Space* se deben configurar los denominados *task* (tareas), las cuales como se nombró anteriormente deben ser definidas con la política de Round Robin (*SCHED_RR*):

Update tasks

Main page | User's defined parameters | Offsets

Main page

Name: no_rt_task

Task type: Periodic

Address space: Space

Processor: mi_processor

Policy: SCHED_RR

Priority: 100

Capacity: 2

Jitter: 0

Deadline: 10

Period: 10

Start time: 0

Blocking time: 0

Criticality: 0

Activation rule:

Stack memory size: 0

Text memory size: 0

Seed: ^ Predictable v Unpredictable

Context switch overhead: 0

Close Cancel Advanced Delete Modify Add

Name	Task type	Processor	Policy	Priority	Capacity	Jitter	Deadline	Period
no_rt	Periodic	mi_processor	RR	100	2	0	10	10
rt_task	Periodic	mi_processor	RR	200	3	0	8	10
rt_task	Periodic	mi_processor	RR	200	5	0	8	10

Update tasks

Main page | User's defined parameters | Offsets

Main page

Name: rt_task_1

Task type: Periodic

Address space: Space

Processor: mi_processor

Policy: SCHED_RR

Priority: 200

Capacity: 3

Jitter: 0

Deadline: 8

Period: 10

Start time: 0

Blocking time: 0

Criticality: 0

Activation rule:

Stack memory size: 0

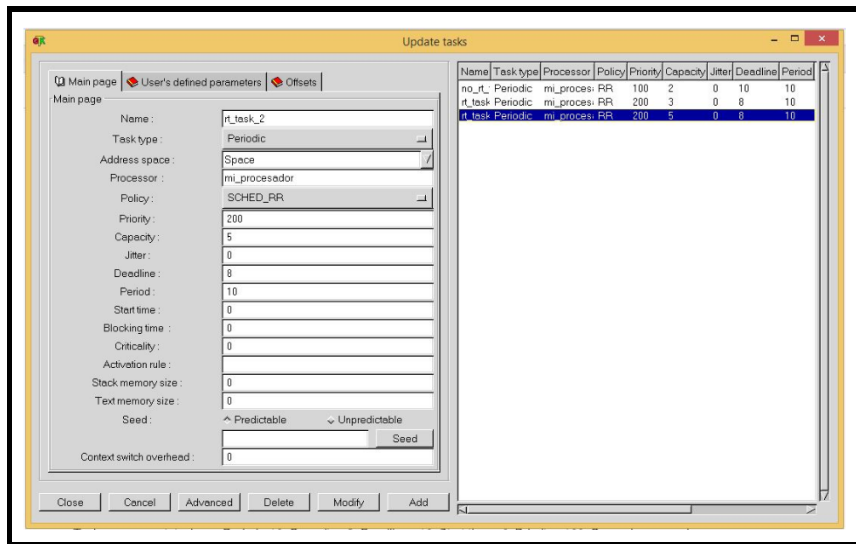
Text memory size: 0

Seed: ^ Predictable v Unpredictable

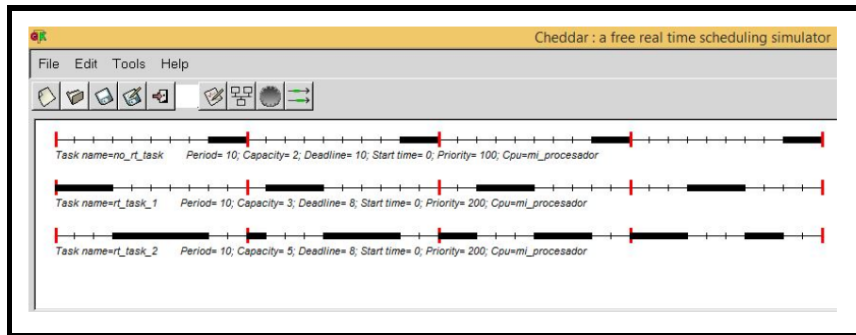
Context switch overhead: 0

Close Cancel Advanced Delete Modify Add

Name	Task type	Processor	Policy	Priority	Capacity	Jitter	Deadline	Period
no_rt	Periodic	mi_processor	RR	100	2	0	10	10
rt_task	Periodic	mi_processor	RR	200	3	0	8	10
rt_task	Periodic	mi_processor	RR	200	5	0	8	10



Como resultado de la parametrización se obtiene la siguiente simulación:



- La parametrización correcta en este caso puede ser la misma que en el inciso 1), pero sin el *Quantum*; también se puede configurar las tareas (*task*) con la política de ejecución FIFO (*First In First Out*, *SCHED_FIFO*). Esto da como resultado que la configuración del procesador y del espacio de memoria van a ser iguales a las descritas en 1), con la única diferencia que el *Quantum* va a ser cero.

Se obtiene el siguiente resultado en la simulación:

