

PARADIGMAS Y LENGUAJES DE
PROGRAMACIÓN
TRABAJO PRÁCTICO NÚMERO 2
– PRÁCTICA –

Ulises C. Ramirez [uli.r19@gmail.com]
Héctor Chripczuk
Verónica Gonzalez

23 de Septiembre, 2018

Versionado

Para el corriente documento se está llevando un versionado a fin de mantener un respaldo del trabajo y además proveer a la cátedra o a cualquier interesado la posibilidad de leer el material en la última versión disponible.

REPOSITORIO: <https://github.com/ulisescolina/UC-PYLP>

–ULISES

Índice de Contenido

1	Definición de conceptos	1
1.1	Granularidad	1
1.1.1	Granularidad fina	1
1.1.2	Granularidad Gruesa	1
2	Grafo de dependencias	1
3	Grado medio de concurrencia	2
4	Ejercicios	2
4.1	Grafos de dependencia 1	2
4.2	Solucion Grafo A	3
4.2.1	Máximo Grado de Concurrencia	3
4.2.2	Camino crítico	3
4.2.3	Grado Medio de Concurrencia	3
4.3	Solucion Grafo B	4
4.3.1	Máximo Grado de Concurrencia	4
4.3.2	Camino crítico	4
4.3.3	Grado Medio de Concurrencia	4
4.4	Grafo de dependencia 2	4
4.5	Solución Consulta 1	5
4.5.1	Camino crítico	5
4.5.2	Grado Medio de Concurrencia	6
4.6	Solución Consulta 2	6
4.6.1	Camino crítico	6
4.6.2	Grado Medio de Concurrencia	6
5	Descomposición	6
6	Asignación, ¿Qué función cumple?	7
7	Ociosidad	7
7.1	¿Qué es la ociosidad?	7
7.2	¿Cuáles son sus principales motivos?	8

8	Estrategías de Asignación	8
9	Grafo de Precedencias	8

1 Definición de conceptos

CONSIGNA: **defina los conceptos de granularidad, granularidad fina y gruesa. ¿Cuáles son las ventajas y desventajas de cada enfoque?**

1.1 Granularidad

Consiste en desglosar, según el grado o nivel que se requiera realizar sobre una tarea de gran tamaño.

1.1.1 Granularidad fina

Consiste en dividir una tarea en N sub-tareas más específicas, presentando mucha comunicación entre estas.

- Ventajas:
 - Permite que se ejecuten de manera paralela las tareas.
- Desventajas:
 - Puede llevar más tiempo en ejecutar todas las tareas, dado a que el nivel de dependencia entre ellas aumenta.

1.1.2 Granularidad Gruesa

Granularidad Gruesa: consiste en dividir una tarea principal en sub-tareas, que engloban varias funciones.

- Ventajas:
 - Permite que la interacción entre diferentes tareas sea mínima, por lo tanto los costos asociados con la comunicacion no son tan dramaticos, ya que la información que se requiere es local, he aquí lo hablado de la localidad de los datos mencionados en clase.
- Desventajas:
 - Las probabilidades de que las tareas se veran afectadas por la secuenciación aumenta.

2 Grafo de dependencias

CONSIGNA: **¿Para qué sirve el grafo de dependencias?**

Este sirve para conocer la relación entre las tareas, donde las aristas conectan una tarea fuente con otra destino, teniendo como característica particular un costes computacional.

3 Grado medio de concurrencia

CONSIGNA: Defina los componentes de la fórmula para calcular el Grado Medio de Concurrencia. ¿Por qué es importante dicha medición?

$$GMC = \frac{\sum_{i=1}^N \text{coste}(\text{nodo}_i)}{L} \quad (1)$$

Se va a dividir la expresión en varias partes para su explicación:

- La función $f(x) = \text{coste}(\text{nodo}_i)$ se puede dividir en dos:
 - nodo_i : es el i -ésimo nodo que se este evaluando en el grafo.
 - $\text{coste}(x)$: es una función cuyo propósito es calcular el coste computacional que tiene el valor x , en este caso es el i -ésimo nodo.
- $\sum_{i=1}^N f(x)$: representa la acumulación de todos los costes computacionales calculados para cada nodo x mediante $f(x)$.
- L : este es el valor acumulado de la suma de los costes de computo asociados a los nodos que componen el grafo del camino crítico.

La importancia de la expresión radica en que permite conocer una estimación de cuantas tareas puede ejecutar mediante un programa que implemente en sus algoritmos una representación de algun grafo.

4 Ejercicios

4.1 Grafos de dependencia 1

CONSIGNAS: calcule el Máximo Grado de Concurrencia, el Camino Crítico, y el Grado Medio de Concurrencia de los grafos presentados en las figuras 1 y 2

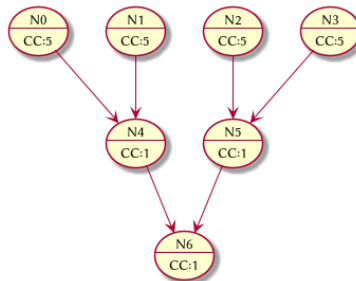


Figure 1: Grafo A

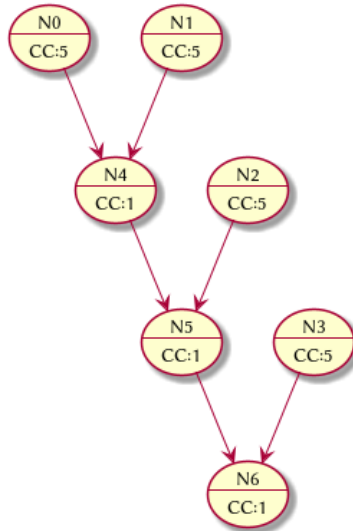


Figure 2: Grafo B

4.2 Solucion Grafo A

4.2.1 Máximo Grado de Concurrencia

Iniciamos con el calculo del Máximo Grado de Concurrencia, el cual en este caso, mediante el analisis del Grafo presentado en la **Figura 1** es de 4

4.2.2 Camino crítico

Mediante el analisis del grafo en la **Figura 1** se procede a identificar el camino critico de forma sencilla este se presenta en la **Figura 3**



Figure 3: Camino crítico para el grafo en la figura 1

4.2.3 Grado Medio de Concurrencia

El valor de $L = 7$ en este caso sale de realizar la suma de los pesos de los nodos que conforman el grafo en la **Figura 3**, se procede a mostrar el Grado Medio de Concurrencia presentado en el **Desarrollo 2**

$$\begin{aligned}
\frac{\sum_{i=0}^N \text{coste}(\text{nodo}_i)}{L} &= \frac{5 + 5 + 5 + 5 + 1 + 1 + 1}{5 + 1 + 1} \\
&= \frac{23}{7} \\
&= 3,2857
\end{aligned} \tag{2}$$

4.3 Solucion Grafo B

4.3.1 Máximo Grado de Concurrencia

Nuevamente procedemos con el calculo del Máximo Grado de Concurrencia, el cual en este caso, mediante el analisis del Grafo presentado en la **Figura 2** también se determina que es de 4

4.3.2 Camino crítico

Mediante el analisis del grafo en la **Figura 2** se procede a identificar el camino critico de la misma manera realizada en el **Punto 4.2** este se presenta en la **Figura 4**

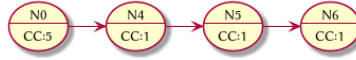


Figure 4: Camino crítico para el grafo en la figura 2

4.3.3 Grado Medio de Concurrencia

El valor de $L = 8$, se llega a la conclusion en del mismo modo que para el ejercicio desarrollado en el **Punto 4.2**. Se procede a mostrar el Grado Medio de Concurrencia en el **Desarrollo 3**

$$\begin{aligned}
\frac{\sum_{i=0}^N \text{coste}(\text{nodo}_i)}{L} &= \frac{5 + 5 + 5 + 5 + 1 + 1 + 1}{5 + 1 + 1 + 1} \\
&= \frac{23}{8} \\
&= 2,875
\end{aligned} \tag{3}$$

4.4 Grafo de dependencia 2

CONSIGNA: se tiene 2 grafos de dependencias **Figura 5** y **Figura 6** que representan 2 posibles soluciones a una consulta sobre una base de datos. Calcule el Grado Medio de Concurrencia de cada grafo de dependencia, siendo el peso de cada nodo igual a la cantidad de registros que recupera.

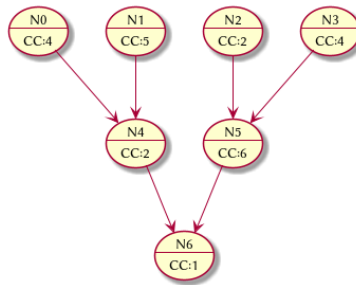


Figure 5: Grafo Dependencias: Consulta 1

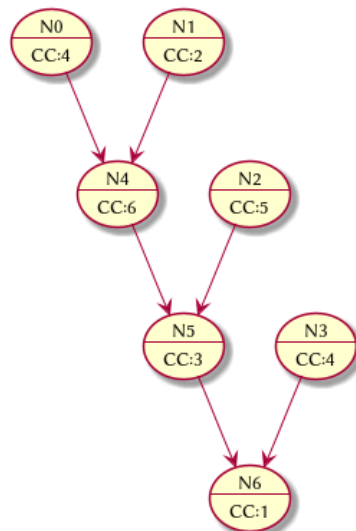


Figure 6: Grafo Dependencias: Consulta 2

4.5 Solución Consulta 1

4.5.1 Camino crítico

Mediante el análisis del grafo en la **Figura 5** se procede a identificar el camino crítico de la misma manera realizada en los puntos anteriores, el grafo correspondiente se presenta en la **Figura 7**



Figure 7: Camino crítico para el grafo en la figura 5

4.5.2 Grado Medio de Concurrencia

El valor de $L = 7$, se llega a la conclusion en del mismo modo que para el ejercicio desarrollado en el **Punto 4.2**, esto es, sumando los pesos de los nodos que conforman el grafo en la **figura 4**, se procede a mostrar el Grado Medio de Concurrencia presentado en el **Desarrollo 4**

$$\begin{aligned} \frac{\sum_{i=0}^N \text{coste}(\text{nodo}_i)}{L} &= \frac{4 + 5 + 2 + 4 + 2 + 6 + 1}{4 + 2 + 1} \\ &= \frac{24}{7} \\ &= 3,4285 \end{aligned} \tag{4}$$

4.6 Solución Consulta 2

4.6.1 Camino crítico

Mediante el analisis del grafo en la **Figura 6** se procede a identificar el camino critico de la misma manera realizada en los puntos anteriores, el grafo correspondiente se presenta en la **Figura 8**

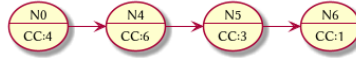


Figure 8: Camino crítico para el grafo en la figura 6

4.6.2 Grado Medio de Concurrencia

El valor de $L = 14$, se llega a la conclusión en del mismo modo que para el ejercicio desarrollado en el **Punto 4.2**, esto es, sumando los pesos de los nodos que conforman el grafo en la **figura 8**, se procede a mostrar el Grado Medio de Concurrencia presentado en el **Desarrollo 5**

$$\begin{aligned} \frac{\sum_{i=0}^N \text{coste}(\text{nodo}_i)}{L} &= \frac{4 + 2 + 5 + 4 + 6 + 3 + 1}{4 + 6 + 3 + 1} \\ &= \frac{24}{14} \\ &= 1,7142 \end{aligned} \tag{5}$$

5 Descomposición

CONSIGNA: ¿Cuál es el objetivo principal de la descomposición? Describa brevemente los tipos de descomposición.

El objetivo principal de la descomposición es *lograr un buen uso de los recursos disponibles* por parte de la aplicación, minimizando el tiempo en el que se realicen operaciones. Los tipos de descomposición se describen a continuación:

- Generales
 - **Descomposición de dominio:** es la división de un programa en sub-divisiones que indican cuales son las funciones que deben contemplar. Es aconsejable utilizarlo cuando la estructura de datos es la más utilizada o la más grande.
 - **Descomposición funcional:** es la división de las funciones de un programa, siendo procesados de manera concurrente. Esta técnica puede ser utilizada siempre y cuando no ocasione excesiva comunicación y/o solapamiento en los datos.
 - **Descomposición recursiva:** basándose en una técnica secuencial divide las funciones de un programa en forma recursiva y combinación de resultados.
- Específicos
 - **Descomposición exploratoria:** se utilizan en problemas de búsqueda, se divide el espacio de búsqueda en partes más pequeñas.
- Mixtos
 - Una combinación de los anteriores.

6 Asignación, ¿Qué función cumple?

La función que cumple, es la de asignar las tareas a cada procesador, minimizando el costo en las comunicaciones y maximizando la utilización de los procesadores involucrados.

7 Ociosidad

CONSIGNA: ¿Qué es la ociosidad? ¿Cuáles son sus principales motivos?

7.1 ¿Qué es la ociosidad?

El término está asociado al tiempo libre que tiene un procesador esperando por realizar alguna tarea.

7.2 ¿Cuáles son sus principales motivos?

Entre los principales motivos para la ociosidad se encuentran:

- **Desequilibrio de carga:** se busca equilibrar tanto los cálculos como las comunicaciones.
- **Espera entre procesos:** esto se da cuando existe una alta interdependencia entre los procesos encargados de realizar una tarea.

8 Estratégias de Asignación

CONSIGNA: **describa brevemente las estrategias generales de asignación.**

Las estrategias generales de asignacion se describen a continuación:

- Agrupamiento:
 - Buscan agrupar un conjunto de tareas para mejorar la localidad.
 - Minimizar el volumen de datos. Esto se puede lograr mediante lo siguiente:
 - * Agrupamiento por bloques de elementos continuos
 - * Agrupar tareas que se comuniquen mucho.
 - * El uso de datos locales para almacenar resultados intermedios.
 - Reduccion de la frecuencia de interacciones. Esto se logra mediante lo siguiente:
 - * Memoria comartida: algoritmos por bloques reduce pérdidas de caché.
 - * Memoria distribuida: agrupar mensajes.
- Replicacion:
 -

9 Grafo de Precedencias

CONSIGNA: **Dado el fragmento de programa concurrente en el Listing 1 obtener el grafo de precedencia asociado.**

Listing 1: Fragmento de programa concurrente

```
1 P0;  
2 COBEGIN  
3   P1;  
4   P2;
```

```

5 COBEGIN
6   P3; P4; P5; P6;
7 COEND
8   P7;
9 COEND
10 P8;

```

A continuación se presenta el grafo de precedencia, la justificación del *por qué* se realizó de esa manera tiene sus bases en dos cuestiones mencionadas en el cursado de la materia.

- La primera de ellas tiene que ver con lo mencionado en [Gortázar Bellas, et al, 2012], y expuesto además en una de las diapositivas de la cátedra y tiene que ver con el orden en el cual ocurre la ejecución de los diferentes procesos que se quiere paralelizar, en resumen, éste acontecimiento es aleatorio, se elige un proceso al azar para que este se ejecute.
- la otra parte, se expuso en clases durante la explicación de **qué es un grafo de dependencias**, en donde se dejó constancia de que para que se pueda iniciar una tarea, debe ejecutarse, en primera instancia, la tarea fuente.

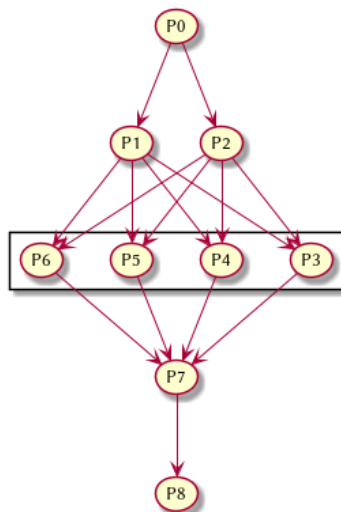


Figure 9: Grafo de precedencia para Listing 1

En este caso tenemos a P1 y P2 como tareas fuente, las cuales deben finalizar, y de ahí se determina cuál es la tarea siguiente, por lo expuesto en el primero de los dos puntos que se expusieron anteriormente, la selección será al azar, por tanto se consideran todos los posibles escenarios de finalización de los padres.

Referencias

- [Gortázar Bellas, et al, 2012] GORTÁZAR BELLAS, FRANCISCO; MARTÍNEZ UNANUE, RAQUEL; FRESNO FERNÁNDEZ, VICTOR. *Lenguajes de Programación y Procesadores - Capítulo 3.5*. Editorial Universitaria Ramon Areces, Madrid, 2012. ISBN: 9788499610702.