

TEMA 1

En todos los casos, responda con claridad y sintéticamente. En los casos que corresponda, NO SE CONSIDERARAN RESPUESTAS SIN JUSTIFICAR. Tiempo Máximo 1h 30 min.

TOTAL = 30 puntos. Para aprobar se requieren 18 puntos, correspondientes al menos a 3 de las partes.

Parte A: Definiciones y preguntas conceptuales que debieran ser contestadas brevemente (10 puntos: Preguntas 1 a 4 un punto c/u, preguntas 5 a 8 1,5 puntos c/u)

- 1- Defina programa concurrente y programa paralelo.
- 2- Defina comunicación entre procesos. Describa los mecanismos de comunicación que conozca.
- 3- Defina propiedad de seguridad y propiedad de vida. Ejemplifique
- 4- ¿En qué consiste la propiedad de “A lo sumo una vez” y qué efecto tiene sobre las sentencias de un programa concurrente?
- 5- Describa el concepto de sincronización barrier y cuál es su utilidad. ¿Qué es una “butterfly barrier”? Ejemplifique gráficamente el funcionamiento para 16 procesos.
- 6- Describa la técnica de *Passing the Baton*. ¿Cuál es su utilidad en la resolución de problemas mediante semáforos? ¿En qué consiste la técnica de *Passing the Condition* y cuál es su utilidad en la resolución de problemas con monitores? ¿Qué relación encuentra entre *passing the condition* y *passing the baton*?
- 7- Caracterice arquitecturas de grano grueso y de grano fino. ¿Cuál es la relación con la granularidad de procesos o aplicaciones?
- 8- ¿En qué consiste la comunicación guardada y cuál es su utilidad? Describa cómo es la ejecución de sentencias de alternativa e iteración que contienen comunicaciones guardadas.

Parte B: Interpretación de código (6 puntos: 2 + 4)

- 9- Dado el siguiente programa concurrente, indique cuál es la respuesta correcta (justifique)

```
int a = 1, b = 0;
```

```
co (await (b = 1) a = 0) // while (a = 1) { b = 1; b = 0; } oc
```

- a) Siempre termina
- b) Nunca termina
- c) Puede terminar o no

- 10- Suponga los siguientes programas concurrentes. Asuma que a es un arreglo bidimensional, EOS es un valor especial que indica el fin de la secuencia de mensajes, y que los procesos son iniciados desde el programa principal.

P 1	chan canal (double) process Genera { int fila, col; double sum; for [fila= 1 to 10000] for [col = 1 to 10000] send canal (a(fila,col)); send canal (EOS) }	process Acumula { double valor, sumT; sumT=0; receive canal (valor); while valor<>EOS { sumT = sumT + valor receive canal (valor); } printf (sumT); }	P 2	chan canal (double) process Genera { int fila, col; double sum; for [fila= 1 to 10000] { sum=0; for [col = 1 to 10000] sum=sum+a(fila,col); send canal (sum); } send canal (EOS) }	process Acumula { double valor, sumT; sumT=0; receive canal (valor); while valor<>EOS { sumT = sumT + valor; receive canal (valor); } printf (sumT); }
--------	--	---	--------	--	--

- a) Qué hacen los programas?
- b) Analice desde el punto de vista del número de mensajes.
- c) Analice desde el punto de vista de la granularidad de los procesos.
- d)Cuál de los programas le parece más adecuado para ejecutar sobre una arquitectura de grano grueso de tipo cluster de PCs? Justifique.

Parte C: Temas para desarrollar (8 puntos: 2 + 3 + 3)

- 11- Compare los algoritmos para resolver el problema de la Sección Crítica (spin locks, tie breaker, ticket, bakery), marcando ventajas y desventajas de cada uno. (NO implemente).
- 12- a) Defina las métricas de speedup y eficiencia. ¿Cuál es el significado de cada una de ellas (qué miden)? ¿Cuál es el rango de valores posibles de cada uno? Ejemplifique.
b) Qué se entiende por escalabilidad de un sistema paralelo?
c) Suponga que la solución a un problema es paralelizada sobre p procesadores de dos maneras diferentes. En un caso, el speedup (S) está regido por la función $S=p-7$ y en el otro por la función $S=p/3$. ¿Cuál de las dos soluciones se comportará más eficientemente al crecer la cantidad de procesadores? Justifique claramente.
d) Suponga que el tiempo de ejecución de un algoritmo secuencial es de 10000 unidades de tiempo, de las cuales sólo el 95% corresponde a código paralelizable. ¿Cuál es el límite en la mejora que puede obtenerse paralelizando el algoritmo? Justifique
- 13- Defina los paradigmas de interacción entre procesos distribuidos heartbeat y servidores replicados. Marque ventajas y desventajas cuando se utiliza comunicación por mensajes sincrónicos o asincrónicos.

Parte D: Ejercicio a resolver (6 puntos)

14- Sea la siguiente solución al problema del producto de matrices de $n \times n$ con P procesos en paralelo con variables compartidas.

```
process worker [w = 1 to P] { # strips en paralelo (p strips de n/P filas)
  int first = (w-1) * n/P + 1    # Primera fila del strip
  int last = first + n/P - 1;    # Última fila del strip
  for [i = first to last] {
    for [j = 1 to n] {
      c[i,j] = 0.0;
      for [k = 1 to n]
        c[i,j] = c[i,j] + a[i,k]*b[k,j];
    }
  }
}
```

Suponga $n=256$ y cada procesador capaz de ejecutar un proceso.

- Calcular cuántas asignaciones, sumas y productos se hacen secuencialmente (caso en que $P=1$), y cuántas se realizan en cada procesador en la solución paralela con $P=8$.
- Si los procesadores P1 a P7 son idénticos, con tiempos de asignación 2, de suma 4 y de producto 6, y si el procesador P8 es 2 veces más lento, calcule cuánto tarda el proceso total concurrente
- ¿Cuál es el valor del speedup?
- ¿Cómo modificaría el código para lograr un mejor speedup?

NOTA: para realizar los cálculos no tenga en cuenta las operaciones de asignación e incremento correspondientes a las sentencias *for*.

Pasos para la entrega del examen.

Debe avisar al docente de qué se va a entregar el examen y luego sacar una foto a cada hoja colocando su DNI a un costado (se recomienda no sacar las fotos en la máxima resolución y combinarlas en un documento PDF).

Enviar las fotos por mail desde una cuenta personal a la cuenta concurrencia.info.unlp@gmail.com. El mail deberá tener en el Asunto el siguiente formato: **apellidoynombre** espacio **legajo** espacio **sala** espacio **númeroDeSala** (por ejemplo: PerezJuan 72603 Sala 1)

La cuenta de la cátedra responderá con una respuesta automática a cada mail recibido que cumpla con el asunto indicado y tenga un archivo adjunto. Es responsabilidad del alumno verificar que los archivos adjuntos no tengan problemas.

TEMA 2

En todos los casos, responda con claridad y sintéticamente. En los casos que corresponda, NO SE CONSIDERARAN RESPUESTAS SIN JUSTIFICAR. Tiempo Máximo 1h 30 min.

TOTAL = 30 puntos. Para aprobar se requieren 18 puntos, correspondientes al menos a 3 de las partes.

Parte A: Definiciones y preguntas conceptuales que debieran ser contestadas brevemente (10 puntos: Preguntas 1 a 4 un punto c/u, preguntas 5 a 8 1,5 puntos c/u)

- 1- Defina programa concurrente y programa distribuido.
- 2- Defina sincronización entre procesos. Describa los mecanismos de sincronización que conozca.
- 3- Defina acciones atómicas condicionales e incondicionales. Ejemplifique.
- 4- ¿En qué consiste la propiedad de “A lo sumo una vez” y qué efecto tiene sobre las sentencias de un programa concurrente?
- 5- ¿Qué significa que un problema sea de “exclusión mutua selectiva” (EMS)? ¿El problema de lectores y escritores es de EMS? ¿Por qué? Si sólo pudiera haber un lector accediendo a la Base de Datos, ¿el problema sería de EMS? ¿Por qué?
- 6- Describa la técnica de *Passing the Baton*. ¿Cuál es su utilidad en la resolución de problemas mediante semáforos? ¿En qué consiste la técnica de *Passing the Condition* y cuál es su utilidad en la resolución de problemas con monitores? ¿Qué relación encuentra entre *passing the condition* y *passing the baton*?
- 7- Caracterice aplicaciones de grano grueso y de grano fino. ¿Cuál es la relación con la granularidad de procesos o aplicaciones?
- 8- ¿En qué consiste la comunicación guardada y cuál es su utilidad? Describa cómo es la ejecución de sentencias de alternativa e iteración que contienen comunicaciones guardadas.

Parte B: Interpretación de código (6 puntos: 2 + 4)

- 9- Dado el siguiente programa concurrente con variables compartidas:

$x = 2; y = 5; z = 8;$

co $x = y - z$ // $y = y * 3x$ // $z = z * 2$ oc

c.1) Cuáles de las asignaciones dentro del co cumplen con la propiedad de “A lo sumo una vez”. Justifique.

c.2) Indique los resultados posibles de la ejecución. Justifique.

Nota 1: las instrucciones NO SON atómicas.

Nota 2: no es necesario que liste TODOS los resultados, pero sí todos los casos que resulten significativos.

- 10- Dado el siguiente bloque de código, indique para cada uno de los ítems si son equivalentes o no. Justificar cada caso (de ser necesario dar ejemplos).

Segmento 1	Segmento 2
<pre> ... int cant=1000; While (true) { IF (cant > 15); datos?(cant) → Sentencias1 □ (cant < 5); datos?(cant) → Sentencias2 □ (INCOGNITA); datos?(cant) → Sentencias3 END IF } ... </pre>	<pre> ... int cant=1000; DO (cant > 15); datos?(cant) → Sentencias1 □ (cant < 5); datos?(cant) → Sentencias2 □ (INCOGNITA); datos?(cant) → Sentencias3 END DO ... </pre>

- a) **INCOGNITA** equivale a: $(cant = 5)$ or $(cant = 15)$
- b) **INCOGNITA** equivale a: $(cant > 0)$
- c) **INCOGNITA** equivale a: $((cant \geq 2) \text{ and } (cant \leq 20))$
- d) **INCOGNITA** equivale a: $((cant > 5) \text{ and } (cant \leq 15))$
- e) **INCOGNITA** equivale a: $((cant > 5) \text{ and } (cant < 15))$

Parte C: Temas para desarrollar (8 puntos: 2 + 3 + 3)

- 11- Compare los algoritmos para resolver el problema de la Sección Crítica (spin locks, tie breaker, ticket, bakery), marcando ventajas y desventajas de cada uno. (NO implemente).
- 12- a) Defina las métricas de speedup y eficiencia. ¿Cuál es el significado de cada una de ellas (qué miden)? ¿Cuál es el rango de valores posibles de cada uno? Ejemplifique.
b) Qué se entiende por escalabilidad de un sistema paralelo?
c) Suponga que la solución a un problema es paralelizada sobre p procesadores de dos maneras diferentes. En un caso, el speedup (S) está regido por la función $S=p-9$ y en el otro por la función $S=p/2$. ¿Cuál de las dos soluciones se comportará más eficientemente al crecer la cantidad de procesadores? Justifique claramente.
d) Suponga que el tiempo de ejecución de un algoritmo secuencial es de 10000 unidades de tiempo, de las cuales sólo el 90% corresponde a código paralelizable. ¿Cuál es el límite en la mejora que puede obtenerse paralelizando el algoritmo? Justifique
- 13- Defina los paradigmas de interacción entre procesos distribuidos prueba-eco y master-worker. Marque ventajas y desventajas cuando se utiliza comunicación por mensajes sincrónicos o asincrónicos.

Parte D: Ejercicio a resolver (6 puntos)

14- Sea la siguiente solución al problema del producto de matrices de $n \times n$ con P procesos en paralelo con variables compartidas.

```
process worker [w = 1 to P] { # strips en paralelo (p strips de n/P filas)
  int first = (w-1) * n/P + 1    # Primera fila del strip
  int last = first + n/P - 1;    # Ultima fila del strip
  for [i = first to last] {
    for [j = 1 to n] {
      c[i,j] = 0.0;
      for [k = 1 to n]
        c[i,j] = c[i,j] + a[i,k]*b[k,j];
    }
  }
}
```

Suponga $n=512$ y cada procesador capaz de ejecutar un proceso.

- Calcular cuántas asignaciones, sumas y productos se hacen secuencialmente (caso en que $P=1$), y cuántas se realizan en cada procesador en la solución paralela con $P=8$.
- Si los procesadores P1 a P7 son idénticos, con tiempos de asignación 1, de suma 2 y de producto 3, y si el procesador P8 es 3 veces más lento, calcule cuánto tarda el proceso total concurrente
- ¿Cuál es el valor del speedup?
- ¿Cómo modificaría el código para lograr un mejor speedup?

NOTA: para realizar los cálculos no tenga en cuenta las operaciones de asignación e incremento correspondientes a las sentencias *for*.

Pasos para la entrega del examen.

Debe avisar al docente de qué se va a entregar el examen y luego sacar una foto a cada hoja colocando su DNI a un costado (se recomienda no sacar las fotos en la máxima resolución y combinarlas en un documento PDF).

Enviar las fotos por mail desde una cuenta personal a la cuenta concurrencia.info.unlp@gmail.com. El mail deberá tener en el Asunto el siguiente formato: **apellidoynombre** espacio **legajo** espacio **sala** espacio **númeroDeSala** (por ejemplo: PerezJuan 72603 Sala 1)

La cuenta de la cátedra responderá con una respuesta automática a cada mail recibido que cumpla con el asunto indicado y tenga un archivo adjunto. Es responsabilidad

TEMA 3

En todos los casos, responda con claridad y sintéticamente. En los casos que corresponda, NO SE CONSIDERARAN RESPUESTAS SIN JUSTIFICAR. Tiempo Máximo 1h 30 min.

TOTAL = 30 puntos. Para aprobar se requieren 18 puntos, correspondientes al menos a 3 de las partes.

Parte A: Definiciones y preguntas conceptuales que debieran ser contestadas brevemente (10 puntos: Preguntas 1 a 4 un punto c/u, preguntas 5 a 8 1,5 puntos c/u)

- 1- Defina programa paralelo y programa distribuido.
- 2- Defina comunicación entre procesos. Describa los mecanismos de comunicación que conozca.
- 3- Defina política de scheduling. Relacione con fairness.
- 4- ¿En qué consiste la propiedad de “A lo sumo una vez” y qué efecto tiene sobre las sentencias de un programa concurrente?
- 5- Explicar en qué consiste el problema de la *sección crítica*, y cuáles son las 4 propiedades que se deben cumplir para resolverlo.
- 6- Describa la técnica de *Passing the Baton*. ¿Cuál es su utilidad en la resolución de problemas mediante semáforos? ¿En qué consiste la técnica de *Passing the Condition* y cuál es su utilidad en la resolución de problemas con monitores? ¿Qué relación encuentra entre *passing the condition* y *passing the baton*?
- 7- Caracterice arquitecturas de grano grueso y de grano fino. ¿Cuál es la relación con la granularidad de procesos o aplicaciones?
- 8- ¿En qué consiste la comunicación guardada y cuál es su utilidad? Describa cómo es la ejecución de sentencias de alternativa e iteración que contienen comunicaciones guardadas.

Parte B: Interpretación de código (6 puntos: 2 + 4)

- 9- Dado el siguiente programa concurrente, indique cuál es la respuesta correcta (justifique)

```
int a = 1, b = 0;
```

```
co (await (b = 1) a = 0) // while (a = 1) { b = 1; b = 0; } oc
```

- a) Siempre termina
- b) Nunca termina
- c) Puede terminar o no

- 10- Suponga los siguientes programas concurrentes. Asuma que *a* es un arreglo bidimensional, *EOS* es un valor especial que indica el fin de la secuencia de mensajes, y que los procesos son iniciados desde el programa principal.

P 1	chan canal (double) process Genera { int fila, col; double sum; for [fila= 1 to 10000] for [col = 1 to 10000] send canal (a(fila,col)); send canal (EOS) }	process Acumula { double valor, sumT; sumT=0; receive canal (valor); while valor<>EOS { sumT = sumT + valor receive canal (valor); } printf (sumT); }	P 2	chan canal (double) process Genera { int fila, col; double sum; for [fila= 1 to 10000] { sum=0; for [col = 1 to 10000] sum=sum+a(fila,col); send canal (sum); } send canal (EOS) }	process Acumula { double valor, sumT; sumT=0; receive canal (valor); while valor<>EOS { sumT = sumT + valor; receive canal (valor); } printf (sumT); }
--------	--	---	--------	--	--

- a) Qué hacen los programas?
- b) Analice desde el punto de vista del número de mensajes.
- c) Analice desde el punto de vista de la granularidad de los procesos.
- d)Cuál de los programas le parece más adecuado para ejecutar sobre una arquitectura de grano grueso de tipo cluster de PCs? Justifique.

Parte C: Temas para desarrollar (8 puntos: 2 + 3 + 3)

- 11- a) Describa brevemente en qué consisten los mecanismos de RPC y Rendezvous. ¿Para qué tipo de problemas son más adecuados?
- b) Por qué es necesario proveer sincronización dentro de los módulos en RPC? ¿Cómo puede realizarse esta sincronización?
- c) Qué elementos de la forma general de rendezvous no se encuentran en el lenguaje ADA?
- 12- a) Defina las métricas de speedup y eficiencia. ¿Cuál es el significado de cada una de ellas (qué miden)? ¿Cuál es el rango de valores posibles de cada uno? Ejemplifique.
- b) Qué se entiende por escalabilidad de un sistema paralelo?
- c) Suponga que la solución a un problema es paralelizada sobre *p* procesadores de dos maneras diferentes. En un caso, el speedup (*S*) está regido por la función $S=p/4$ y en el otro por la función $S=p-5$. ¿Cuál de las dos soluciones se comportará más eficientemente al crecer la cantidad de procesadores? Justifique claramente.
- d) Suponga que el tiempo de ejecución de un algoritmo secuencial es de 1000 unidades de tiempo, de las cuales sólo el 95% corresponde a código paralelizable. ¿Cuál es el límite en la mejora que puede obtenerse paralelizando el algoritmo? Justifique
- 13- Defina los paradigmas de interacción entre procesos servidores replicados y token passing. Marque ventajas y desventajas cuando se utiliza comunicación por mensajes sincrónicos o asincrónicos.

Parte D: Ejercicio a resolver (6 puntos)

14- Sea la siguiente solución al problema del producto de matrices de $n \times n$ con P procesos en paralelo con variables compartidas.

```
process worker [w = 1 to P] { # strips en paralelo (p strips de n/P filas)
  int first = (w-1) * n/P + 1    # Primera fila del strip
  int last = first + n/P - 1;    # Ultima fila del strip
  for [i = first to last] {
    for [j = 1 to n] {
      c[i,j] = 0.0;
      for [k = 1 to n]
        c[i,j] = c[i,j] + a[i,k]*b[k,j];
    }
  }
}
```

Suponga $n=1024$ y cada procesador capaz de ejecutar un proceso.

- a) Calcular cuántas asignaciones, sumas y productos se hacen secuencialmente (caso en que $P=1$), y cuántas se realizan en cada procesador en la solución paralela con $P=8$.
- b) Si los procesadores P1 a P7 son idénticos, con tiempos de asignación 1, de suma 2 y de producto 3, y si el procesador P8 es 2 veces más lento, calcule cuánto tarda el proceso total concurrente
- c) ¿Cuál es el valor del speedup?
- d) ¿Cómo modificaría el código para lograr un mejor speedup?

NOTA: para realizar los cálculos no tenga en cuenta las operaciones de asignación e incremento correspondientes a las sentencias *for*.

Pasos para la entrega del examen.

Debe avisar al docente de qué se va a entregar el examen y luego sacar una foto a cada hoja colocando su DNI a un costado (se recomienda no sacar las fotos en la máxima resolución y combinarlas en un documento PDF).

Enviar las fotos por mail desde una cuenta personal a la cuenta concurrencia.info.unlp@gmail.com. El mail deberá tener en el Asunto el siguiente formato: **apellidoynombre** espacio **legajo** espacio **sala** espacio **númeroDeSala** (por ejemplo: PerezJuan 72603 Sala 1)

La cuenta de la cátedra responderá con una respuesta automática a cada mail recibido que cumpla con el asunto indicado y tenga un archivo adjunto. Es responsabilidad

TEMA 4

En todos los casos, responda con claridad y sintéticamente. En los casos que corresponda, NO SE CONSIDERARAN RESPUESTAS SIN JUSTIFICAR. Tiempo Máximo 1h 30 min.

TOTAL = 30 puntos. Para aprobar se requieren 18 puntos, correspondientes al menos a 3 de las partes.

Parte A: Definiciones y preguntas conceptuales que debieran ser contestadas brevemente (10 puntos: Preguntas 1 a 4 un punto c/u, preguntas 5 a 8 1,5 puntos c/u)

- 1- Defina programa distribuido y programa paralelo.
- 2- Defina sincronización entre procesos. Describa los mecanismos de sincronización que conozca.
- 3- Defina la instrucción Test & Set. ¿Cuál es su utilidad para la sincronización?
- 4- ¿En qué consiste la propiedad de “A lo sumo una vez” y qué efecto tiene sobre las sentencias de un programa concurrente?
- 5- ¿Qué significa que un problema sea de “exclusión mutua selectiva” (EMS)? Describa el problema de los 5 filósofos. ¿Este problema es de EMS? ¿Por qué? Si sólo hubiera 3 filósofos, ¿el problema sería de EMS? ¿Por qué?
- 6- Describa la técnica de *Passing the Baton*. ¿Cuál es su utilidad en la resolución de problemas mediante semáforos? ¿En qué consiste la técnica de *Passing the Condition* y cuál es su utilidad en la resolución de problemas con monitores? ¿Qué relación encuentra entre *passing the condition* y *passing the baton*?
- 7- Caracterice aplicaciones de grano grueso y de grano fino. ¿Cuál es la relación con la granularidad de procesos o aplicaciones?
- 8- ¿En qué consiste la comunicación guardada y cuál es su utilidad? Describa cómo es la ejecución de sentencias de alternativa e iteración que contienen comunicaciones guardadas.

Parte B: Interpretación de código (6 puntos: 2 + 4)

- 9- Dado el siguiente programa concurrente con variables compartidas:

$x = 2; y = 5; z = 8;$

co $x = y - z$ // $y = y * 3x$ // $z = z * 2$ oc

- c.1) Cuáles de las asignaciones dentro del co cumplen con la propiedad de “A lo sumo una vez”. Justifique.
- c.2) Indique los resultados posibles de la ejecución. Justifique.

Nota 1: las instrucciones NO SON atómicas.

Nota 2: no es necesario que liste TODOS los resultados, pero sí todos los casos que resulten significativos.

- 10- Dado el siguiente bloque de código, indique para cada uno de los ítems si son equivalentes o no. Justificar cada caso (de ser necesario dar ejemplos).

Segmento 1	Segmento 2
<pre>... int cant=1000; While (true) { IF (cant > 15); datos?(cant) → Sentencias1 □ (cant < 5); datos?(cant) → Sentencias2 □ (INCOGNITA); datos?(cant) → Sentencias3 END IF } ...</pre>	<pre>... int cant=1000; DO (cant > 15); datos?(cant) → Sentencias1 □ (cant < 5); datos?(cant) → Sentencias2 □ (INCOGNITA); datos?(cant) → Sentencias3 END DO ...</pre>

- a) **INCOGNITA** equivale a: **(cant = 5) or (cant = 15)**
- b) **INCOGNITA** equivale a: **(cant > 0)**
- c) **INCOGNITA** equivale a: **((cant >= 2) and (cant <= 20))**
- d) **INCOGNITA** equivale a: **((cant > 5) and (cant <= 15))**
- e) **INCOGNITA** equivale a: **((cant > 5) and (cant < 15))**

Parte C: Temas para desarrollar (8 puntos: 2 + 3 + 3)

- 11- a) Describa brevemente en qué consisten los mecanismos de RPC y Rendezvous. ¿Para qué tipo de problemas son más adecuados?
- b) Por qué es necesario proveer sincronización dentro de los módulos en RPC? ¿Cómo puede realizarse esta sincronización?
- c) Qué elementos de la forma general de rendezvous no se encuentran en el lenguaje ADA?
- 12- a) Defina las métricas de speedup y eficiencia. ¿Cuál es el significado de cada una de ellas (qué miden)? ¿Cuál es el rango de valores posibles de cada uno? Ejemplifique.
- b) Qué se entiende por escalabilidad de un sistema paralelo?
- c) Suponga que la solución a un problema es paralelizada sobre p procesadores de dos maneras diferentes. En un caso, el speedup (S) está regido por la función $S=p/2$ y en el otro por la función $S=p-10$. ¿Cuál de las dos soluciones se comportará más eficientemente al crecer la cantidad de procesadores? Justifique claramente.
- d) Suponga que el tiempo de ejecución de un algoritmo secuencial es de 1000 unidades de tiempo, de las cuales sólo el 90% corresponde a código paralelizable. ¿Cuál es el límite en la mejora que puede obtenerse paralelizando el algoritmo? Justifique
- 13- Defina los paradigmas de interacción entre procesos distribuidos heartbeat, y token passing. Marque ventajas y desventajas cuando se utiliza comunicación por mensajes sincrónicos o asincrónicos.

Parte D: Ejercicio a resolver (6 puntos)

14- Sea la siguiente solución al problema del producto de matrices de $n \times n$ con P procesos en paralelo con variables compartidas.

```
process worker [w = 1 to P] { # strips en paralelo (p strips de n/P filas)
  int first = (w-1) * n/P + 1    # Primera fila del strip
  int last = first + n/P - 1;    # Ultima fila del strip
  for [i = first to last] {
    for [j = 1 to n] {
      c[i,j] = 0.0;
      for [k = 1 to n]
        c[i,j] = c[i,j] + a[i,k]*b[k,j];
    }
  }
}
```

Suponga $n=1024$ y cada procesador capaz de ejecutar un proceso.

- a) Calcular cuántas asignaciones, sumas y productos se hacen secuencialmente (caso en que $P=1$), y cuántas se realizan en cada procesador en la solución paralela con $P=8$.
- b) Si los procesadores P1 a P7 son idénticos, con tiempos de asignación 2, de suma 4 y de producto 6, y si el procesador P8 es 3 veces más lento, calcule cuánto tarda el proceso total concurrente
- c) ¿Cuál es el valor del speedup?
- d) ¿Cómo modificaría el código para lograr un mejor speedup?

NOTA: para realizar los cálculos no tenga en cuenta las operaciones de asignación e incremento correspondientes a las sentencias *for*.

Pasos para la entrega del examen.

Debe avisar al docente de qué se va a entregar el examen y luego sacar una foto a cada hoja colocando su DNI a un costado (se recomienda no sacar las fotos en la máxima resolución y combinarlas en un documento PDF).

Enviar las fotos por mail desde una cuenta personal a la cuenta concurrency.info.unlp@gmail.com. El mail deberá tener en el Asunto el siguiente formato: **apellidoynombre** espacio **legajo** espacio **sala** espacio **númeroDeSala** (por ejemplo: PerezJuan 72603 Sala 1)

La cuenta de la cátedra responderá con una respuesta automática a cada mail recibido que cumpla con el asunto indicado y tenga un archivo adjunto. Es responsabilidad