

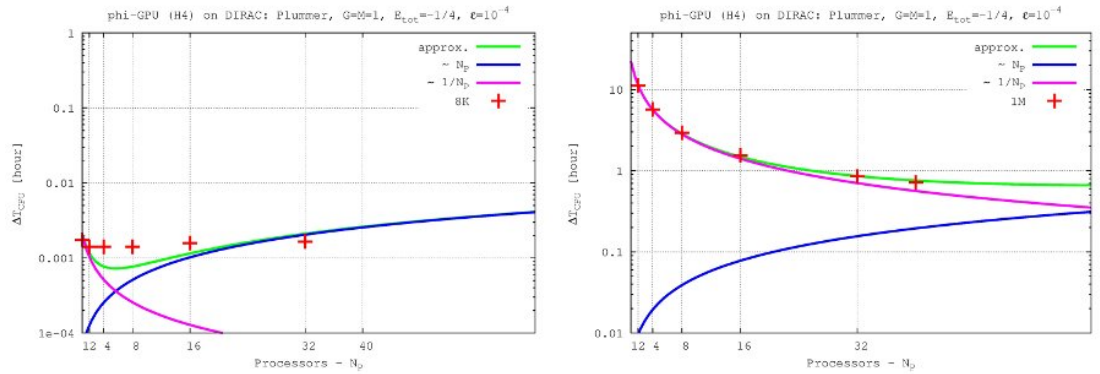
# Programación Paralela

## Exámen Parcial

**Prof. José Fiestas**  
jose.fiestas@uni.edu.pe  
Universidad Nacional de Ingeniería

- 
- El exámen dura 100 minutos
  - Está permitido consulta de material de clase, pero ningún otro material
  - La entrega del exámen se hará en Classroom. Subir la solución del exámen en formato pdf o foto
- 

1) Las siguientes gráficas representan tiempo vs. número de nodos de un código de N-cuerpos (izquierda: 8K cuerpos, derecha: un millón de cuerpos), señale que representan las curvas y su punto de cruce en cada caso ¿Cómo sería una grafica para  $N = 250K$ ?  
Diagrame velocidad vs. nodos, equivalente a los graficos mostrados y relacione ambos. (4 puntos)



2) Un código de N-cuerpos llega a una velocidad teórica en FLOPs, de aproximadamente

$$P \approx \frac{\gamma N^{2+x}}{\alpha N^{2+x}/np + \beta * \log np} \quad (1)$$

donde  $\gamma = 500$  es el número de operaciones de coma flotante (FLOP) por partícula, por unidad de tiempo,  $\alpha = 10^{-9}$ , y  $\beta = 1$  son constantes de hardware,  $np$  es el número de procesos, y  $x = 0.31$  es una constante experimental. Calcule los FLOPs teóricos de una aplicación de N-cuerpos en un supercomputador de 5000 nodos.

Realice un análisis de escalabilidad, y determine si representa un caso de strong o weak scaling **(3 puntos)**

**3)** Haga el análisis de complejidad del algoritmo quicksort mostrado, en el cual cada elemento de la lista es asignado a un proceso

---

```

procedure theoQSort(d,i,p)
  if (p=1) return;
  j := common random element from 0..p-1 for partition;
  v := d@j; //Pivot
  f := d ≤ v;
  j :=  $\sum_{k=0}^i f@k$ ; //Präfixsumme
  p' := j@(p-1);
  if (f) send d to PE j-1;

  else send d to PE p'+i-j;
  receive d;
  if (i < p') {
    join partition "left";
    theoQSort(d,i,p');
  } else {
    join partition "right";
    theoQSort(d,i-s,p-p');
  }

```

---

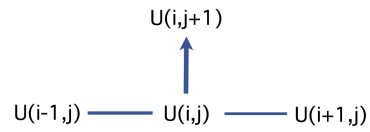
Ahora extienda el concepto para un  $n \gg 1$ , i.e.  $p < n$   
**(4 puntos)**

**4)** Desarrolle un algoritmo PRAM para el siguiente procedimiento en una red de procesos:

Cada proceso envía un mensaje a otro proceso, escogido en forma aleatoria. Este último vuelve a enviar el mensaje de retorno, al mismo proceso original. En el momento en que el primer proceso recibe la señal de confirmación, envía un mensaje a otro proceso, escogido en forma aleatoria. De esta forma, todos los  $p$  procesos deben enviar  $n$  mensajes (y esperar la señal de confirmación).

Cuando el procedimiento ha terminado, el proceso 0 determina el tiempo que ha tomado el procedimiento de envío y recibo de, un total de  $2np$  mensajes. **(4 puntos)**

5) Un algoritmo de suavizado en procesamiento gráfico utiliza los valores de los pixels vecinos para modificar el valor de cada pixel de la imagen, de acuerdo al siguiente gráfico



Formule un algoritmo PRAM para particionar el dominio y resolver el problema de comunicación entre procesos, si cada proceso recibe  $n/p$  pixels. Asuma que la operación de suavizado tiene  $T = O(1)$  (**5 puntos**)