

# PRÁCTICA 1: MPI

## Busca aleatoria cadena “secreta”

Se desea realizar un programa que permita encontrar una palabra “secreta” de una longitud determinada (con un conjunto de caracteres determinado) generando combinaciones aleatorias, de manera que, se comprueben y los caracteres acertados no se vuelvan a generar. Para ello habrá tres tipos de procesos: E/S, Comprobadores y Generadores

- **Proceso de E/S:**
  - o Es el que conocerá la palabra, el número de procesos Comprobadores y, por lo tanto, el de Generadores
  - o Irá recibiendo las “pistas” (cadena parcialmente descubierta) de los Generadores y, si está activado el modo pista, se lo notificará al resto de Generadores.
- **Procesos Comprobadores:**
  - o El proceso de E/S les informará de la palabra “secreta”.
  - o Los procesos Generadores le mandarán una palabra para comprobarla. Devolverá al Generador la palabra con los caracteres acertados.
- **Procesos Generadores:**
  - o El proceso de E/S les informará de la longitud de la palabra secreta y ellos generarán una combinación aleatoria y consultarán al proceso Comprobador.
    - El resultado de la consulta es una cadena con la misma longitud que la enviada que contiene los caracteres acertados, de tal manera que lo almacenarán para que, a la hora de generar otra combinación, no generar las posiciones acertadas.
    - Si la respuesta tiene nuevos caracteres acertados mandarán la cadena al proceso de E/S, y si está activado el modo pista, este se lo notificará al resto de procesos Generadores para que no generen esos caracteres.

```

mpirun -np 7 -oversubscribe mpi 2 1
NUMERO DE PROCESOS: Total 7: E/S: 1, Comprobadores: 2, Generadores: 4

NOTIFICACION TIPO
01) 0
02) 0
03) 1
04) 2
05) 1
06) 2

NOTIFICACION PALABRA COMPROBADORES
01) PRACTICAMPI2122, 16
02) PRACTICAMPI2122, 16

BUSCANDO

06) PISTA.....:      A
03) PISTA.....:      I
06) PISTA.....:      I A I
05) PISTA.....:      A PI
03) PISTA.....: P      A I
03) PISTA.....: P      I AMPI
05) PISTA.....: P      TI AMPI
06) PISTA.....: P      I AMPI 2
06) PISTA.....: P      TICAMPI 2
03) PISTA.....: P      TICAMPI 12
06) PISTA.....: PR      TICAMPI 12
05) PISTA.....: PR      CTICAMPI 12
06) PISTA.....: PR      CTICAMPI 122
06) PISTA.....: PR      CTICAMPI2122

PALABRA ENCONTRADA POR 6
BUSCADA...: PRACTICAMPI2122
ENCONTRADA: PRACTICAMPI2122

```

### SE PIDE:

- **Realizar un programa en MPI que permita distribuir el cálculo en N procesos.** Para ello habrá:
  - o *Un proceso encargado de la E/S que:*
    - Será el conocedor de la palabra a descubrir
    - Notificará la longitud al resto y adicionalmente, a los procesos Comprobadores la palabra:
      - La palabra tendrá un conjunto acotado de posibles caracteres, por ejemplo:
        - o "ABCDEFGHIIJKLMNIOQUJAKSGHAJHGFSGfghafsgghafstjharfasdhjshdjh216531823612678&&!1281asdasdjahsdkjhadsljhaejuj"

- Tanto el rango, como el carácter que marcará como no coincidente, se deben definir también, para que los procesos Generadores y Comprobadores puedan trabajar, por ejemplo:
    - #define CHAR\_NF 32 // Para marcar no encontrado (espacio, por simplificar)
    - #define CHAR\_MAX127
    - #define CHAR\_MIN33
  - Será el conocedor de cuantos procesos Comprobadores existen y si es posible seguir adelante con el número de procesos lanzados
  - Dependiendo del numero de procesos lanzados y el de Comprobadores, habrá un número de procesos Generadores.
    - Debe indicar al resto de procesos si son Comprobadores o si son Generadores (y dependiendo de cómo se hagan las asignaciones, cuál es su Comprobador)
  - También sabrá si se usan pistas o no.
  - Una vez notificados el tipo de proceso, y la longitud de la palabra (y la palabra a los Comprobadores) según cada caso, se quedará a la espera de recibir de los Generadores:
    - Una cadena con los caracteres ya encontrados
      - Que si tiene activo el modo “pista” lo distribuye al resto de Generadores
    - La palabra ya encontrada
  - Una vez encontrada la palabra mandará terminar a los Comprobadores y Generadores y recibirá las estadísticas pertinentes.
- **Resto de procesos:**
  - Recibirán la longitud de la palabra.
  - Recibirán de alguna manera una indicación de si son Comprobadores o Generadores, y en caso de ser Generadores quien es su Comprobador.
  - **Procesos Comprobadores:**
    - Recibirán la palabra a comprobar.
    - Se quedarán a la espera de:
      - Que el proceso E/S les mande acabar
      - Un proceso Generador les mande una palabra
        - Comprobarán la palabra
        - **Forzarán una espera para dar peso al cálculo**
        - Devolverán la respuesta con los caracteres acertados en sus posiciones.
    - Una vez le hayan mandado acabar mandará las estadísticas tomadas al proceso de E/S
  - **Procesos Generadores:**
    - Deberán tener en cuenta que:
      - El proceso E/S les puede mandar acabar
      - El proceso E/S les puede mandar una pista (cadena con los caracteres encontrados por otros) y actualizaran su cadena de caracteres encontrados.
    - Generará una palabra de la longitud indicada:
      - Deben tener en cuenta los caracteres ya detectados
      - Generarán de manera aleatoria los caracteres en los rangos determinados (**cuidado con la semilla**)
      - **Forzará una espera para dar peso al cálculo**
      - Mandarán la consulta a su proceso Comprobador
      - Recibirán la respuesta y actualizarán su cadena de caracteres encontrados
    - Una vez les hayan mandado acabar mandarán las estadísticas tomadas al proceso de E/S
- **Una vez realizado el programa realizar un estudio de rendimiento en el cual se vea que:**
  - Aumentando el número de procesos (y máquinas si es posible) se reduce el tiempo de cálculo, o no, hasta un cierto momento.
    - Habrá que tener en cuenta que se necesitan mínimo 3 procesos
    - Se forzarán esperas para dar peso al cálculo.
 

```
#define PESO_COMPROBAR          5000000
#define PESO_GENERAR            10000000
void fuerza_espera(unsigned long peso)
{
    for (unsigned long i=1; i<1*peso; i++) sqrt(i);
}
```
    - **Se debe partir de tiempos de cálculo altos para ver el efecto.**
    - Los procesos Comprobadores y Generadores deberán tomar datos de cuántas cadenas comprueban-generan, cuanto tiempo dedican a su tarea y cuanto a esperar (la espera forzada entra dentro de la tarea), etc...
    - Se mostrarán datos que nos permitan analizar los resultados:
      - Porcentajes de tiempos ocupados o esperando
      - Comprobaciones por segundo

- **Analizar los datos obtenidos y problemas que pueden surgir, así como proponer mejoras a partir de los resultados obtenidos.**

**SE DEBERÁ ENTREGAR:**

- Código fuente del programa realizado.
- Presentación empleada en el Seminario
  - Debe de quedar plasmado en la presentación de manera explícita los resultados y conclusiones obtenidas.
  - Es bueno realizar y mostrar gráficas comparativas

**NOTA:**

- La práctica se realizará en grupos de 4 personas
- Se deberá subir al Studium en un único archivo en la fecha marcada por un único componente del grupo.
- La presentación de la práctica se realizará en la sesión marcada como Seminario y se entregará el mismo día de la presentación.
- **La detección de copia parcial o total de la práctica conllevará la suspensión de las prácticas, y por tanto de la asignatura.**

**NUMERO DE PROCESOS:** Total 8: E/S: 1, Comprobadores: 2, Generadores: 5

## NOTIFICACION TIPO

01)	0
02)	0
03)	1
04)	2
05)	1
06)	2
07)	1

## NOTIFICACION PALABRA COMPROBADORES

01) ABCDEFGHIJ KLMNIOQUJAKSGHAJHGFSGfghafsgfhafsgfhafstjharfasdhjshdjh216531823612678&&|1281asdasdjhaskdjahdsjljaejuj, 111  
02) ABCDEFGHIJ KLMNIOQUJAKSGHAJHGFSGfghafsgfhafsgfhafstjharfasdhjshdjh216531823612678&&|1281asdasdjhaskdjahdsjljaejuj, 111

**BUSCANDO**

[illegible]

## PALABRA ENCONTRADA POR 6

BUSCADA: ABCDEFGHIJ KLMNIOQUAKSGHAJHGFSGfghafsgfhafsgfhafstjgharfasdjhshdjh216531823612678&&11281asdasdajhsdkjhads1jhjaejuj  
ENCENTRADA: ABCDEFGHIJ KLMNIOQUAKSGHAJHGFSGfghafsgfhafsgfhafstjgharfasdjhshdjh216531823612678&&11281asdasdajhsdkjhads1jhjaejuj  
03) Fin Generadores  
04) Fin Generadores

05) Fin Generadores  
 06) Fin Generadores  
 07) Fin Generadores

**Generadores**

Iteraciones Total TpoTotal TpoGen TpoEspComp  
 03) 81 81 | 13.222131, 8.604991 (65.080214), 4.616749 (34.916831)  
 04) 84 165 | 13.193625, 8.806322 (66.746797), 4.386898 (33.250074)  
 05) 81 246 | 13.170769, 8.652176 (65.692261), 4.518141 (34.304309)  
 06) 82 328 | 13.139981, 8.721779 (66.375890), 4.417740 (33.620594)  
 07) 82 410 | 13.276351, 8.643174 (65.102029), 4.632856 (34.895550)  
 01) Fin Comprobadores  
 02) Fin Comprobadores

**Comprobadores**

Iteraciones IteracionesTotal TpoTotal TpoComp  
 01) 244 244 | 13.276474 12.851653 (96.800198)  
 02) 166 410 | 13.276629 8.685034 (65.415956)  
 >-----ESTADÍSTICAS TOTALES-----<  
 Numero Procesos: 8  
 Tiempo Procesamiento: 13.276890

>-----FIN-----<

Esto solo es un ejemplo de posibles resultados

Nº Procesos	E/S	Comprobadores	Generadores	Pistas	Comprobaciones	Tpo	G1	G2	G3	G1		G2		G3		G4		G5		G6		TOTAL
							%TC	%TC	%TC	%TG	%TE	%TG	%TE	%TG	%TE	%TG	%TE	%TG	%TE	%TG	%TE	
3	1	1	1	1	420	51.32	33.33			66.66	33.33											8.09
4	1	1	2	1	520	32.72	66.66			66.08	33.87	67.01	32.95									15.89
5	1	1	3	1	432	21.75	93.26			61.15	38.85	61.57	38.42	61.32	38.67							19.86
6	1	1	4	1	718	33.72	99.75			54.88	45.12	50.41	49.59	51.09	48.89	53.37	46.63					21.29
7	1	1	5	1	498	26.31	99.68			38.69	61.29	37.49	60.51	33.18	60.82	37.38	62.61	40.5	59.49			18.33
8	1	1	6	1	420	22.27	99.55			30.05	69.94	30.11	69.89	30.07	69.93	47.38	52.02	30.16	69.82	29.37	69.39	18.86
3	1	2	0	1	X																	0.00
4	1	2	1	1	582	72.15	33.29	0		66.7	33.29											8.07
5	1	2	2	1	518	36.01	34.18	34.06		65.74	34.25	65.94	34.06									14.38
6	1	2	3	1	494	24.81	65.51	32.34		65.17	34.83	67.52	32.47	65.67	34.32							19.31
7	1	2	4	1	562	21.94	66.21	63.48		66.59	33.41	67.97	32.03	58.09	41.91	66.38	33.62					25.62
8	1	2	5	1	410	13.28	96.8	65.42		65.08	34.92	66.75	34.3	65.69	34.3	66.38	33.62	65.1	34.89			30.87
3	1	3	0	1	X																	0.00
4	1	3	0	1	X																	0.00
5	1	3	1	1	468	67.69	33.06	0	0	66.33	33.07											6.31
6	1	3	2	1	310	22.89	33.73	32.95	0	67.27	32.73	67.01	32.99									13.54
7	1	3	3	1	625	31.99	31.66	32.05	33.19	68.34	31.66	67.91	32.09	66.73	33.26							19.54
8	1	3	4	1	557	22.19	66.17	32.66	33.47	66.51	33.49	67.15	32.85	66.51	33.48	66.07	33.93					25.10
8	1	3	4	0	1827	72.45	66.48	33.15	33.43	66.36	33.63	66.74	33.26	66.51	33.49	66.37	33.62					25.22