

# Proyecto de segundo parcial

December 2, 2016



# Proyecto de segundo parcial

- Usando algoritmos genéticos, resolver el problema del agente viajero para  $N$  ciudades ( $5 \leq N \leq 40$ ).
- El programa deberá leer una tabla de distancias desde un archivo de texto.
- Se deberá usar paso de mensajes para poder distribuir el procesamiento de la función objetivo en los diferentes nodos - asuma que deberá poder ejecutar el programa en entre 4 y 32 nodos.
- Pueden formar equipos de máximo 2 personas.

# Archivo de datos de entrada y configuración

- Asuma que siempre será un archivo que contendrá información de una matriz simétrica (para facilitar las cosas).
- Puede leer la matriz desde un archivo de texto, el contenido del archivo es a su criterio, como sugerencias, pueden contener:
  - ▶ Número de ciudades
  - ▶ Distribución del procesamiento en los nodos - en cada iteración de la población, cuántos elementos de la población procesará.
  - ▶ Estrategia de selección: rank, torneo, aleatoria.
  - ▶ Nombre de archivo para generar los reportes.
  - ▶ Si usará elitismo - cuántos elementos de la población entre los más aptos deben mantenerse hacia la siguiente generación.
  - ▶ Probabilidad de mutación y número de elementos a mutar por generación.

	0	1	2
0	0	20	inf
1	20	0	30
2	inf	30	0

# Archivos de salida

- Deberá generar un archivo con formato para matlab (.m) que pueda ejecutarse en octave/matlab para mostrar una gráfica que indique el mejor valor de cada generación.
- Deberá calcular el tiempo de ejecución total usando funciones de MPI u openMP (*MPI\_Wtime()*, *omp\_get\_wtime()*).

# Entregables

- Código fuente en C + OpenMP + MPI.
- El código debe estar documentado.
- Prueba física de ejecución con una cantidad de entre 4 y 32 procesadores.
- Documento con una introducción breve a los algoritmos genéticos y el problema que están resolviendo (una cuartilla como máximo). Agregar tablas de comparación y gráficas en tiempo de ejecución del algoritmo variando:
  - ▶ Número de procesadores distribuidos usados (4, 16, 32).
  - ▶ Número de elementos de la población (100, 1000, 10000, 50000) contra número de generaciones (10, 25, 50) → una tabla para cada cantidad de procesadores diferente - es decir 3 tablas.
  - ▶ Para esta tabla reporten el algoritmo únicamente con el tamaño más grande de ciudades que hayan probado.
  - ▶ Introduzcan las graficas generadas por el programa y matlab donde se muestre la convergencia hacia la mejor solución de su algoritmo → para los tres métodos distintos de selección, únicamente para la población más grande con el número de procesadores más grande que hayan usado.
  - ▶ Todas las gráficas deberán tener su interpretación.
- Sus conclusiones.