

Reprograme la versión secuencial de la función *viajante* de la tarea 7, de manera que si el usuario aborta con control-C (señal SIGINT), se entregue la mejor solución encontrada hasta el momento. Debe usar estas 2 metodologías para resolver el problema.

**Metodología A:** Modifique *viajante* estableciendo una rutina de atención para la señal SIGINT. Cuando se gatille esta señal, y por lo tanto se invoque su rutina de atención, coloque una marca en una variable global que indique el evento. Modifique la condición del ciclo principal de *viajante* para que consulte la variable global y termine el ciclo cuando la variable global indique que se gatilló SIGINT. Entregue la mejor solución calculada hasta el momento. No olvide colocar el atributo *volatile* en la variable global.

**Metodología B:** Modifique *viajante* marcando un punto de retorno con *setjmp* al inicio y establezca una rutina de atención para la señal SIGINT. Cuando se invoque la rutina de atención, use *longjmp* para escapar del ciclo de generación de soluciones aleatorias yendo directamente al punto de retorno marcado al inicio. Entregue la mejor solución calculada hasta el momento. Explique en un comentario al inicio de su archivo por qué esta metodología podría entregar un resultado incorrecto. Es decir la distancia mínima retornada por *viajante* podría no corresponder a la distancia recorrida por la ruta entregada en el arreglo *z*.

**Importante:** Para poder hacer esta tarea Ud. debe haber estudiado previamente las cátedras del martes 15 (*setjmp/longjmp*) y jueves 17 (señales). En esta última, se usaron las metodologías A y B para resolver el problema de la búsqueda de un factor.

### Instrucciones

Baje *t8.zip* de U-cursos y descomprímalo. El directorio contiene los archivos *test-t8.c*, *Makefile*, *viajante.h* (con los encabezados requeridos), *t8a.c* y *t8b.c*. Sin hacer ningún cambio en estos archivos, compile con el comando *make* (sin parámetros) y ejecute con *./test-t8a* o *./test-t8b*. Esta es la salida de ambos programas:

```
Test 1: ejemplo con solo 6 ciudades
Calculando secuencialmente
Resultado final: nperm=500000000 distancia minima=2.113459
Ruta: 0 1 3 4 5 2 0
Test 1 aprobado
```

```
-----
Test 2: Un ejemplo grande con 33 ciudades
Calculando secuencialmente
Resultado final: nperm=100000000 distancia minima=10.715848
Ruta: 0 18 3 23 11 7 22 4 6 16 5 12 24 15 9 17 30 27 14 2
28 26 20 1 31 25 13 32 8 21 29 10 19 0
Test 2 aprobado
```

Programe la metodología A en el archivo *t8a.c* y la metodología B en *t8b.c*, y compile con *make*. Pruebe su tarea de la siguiente manera. Ejecute *./test-t8a* o *./test-t8b* sin ingresar control-C. La salida debe ser la misma del cuadro de arriba. Vuelva a ejecutar nuevamente ambos programas, pero ingrese control-C en cuanto aparezca el texto “calculando ...” asociado al test 1. Sin demora debería aparecer la mejor distancia calculada hasta el momento. Es normal que el programa reclame que el resultado es insuficiente. Vuelva a hacer lo mismo presionando control-C en cuanto aparezca el texto “calculando ...” asociado al test 2. Lo importante en esta tarea es que el resultado debe aparecer inmediatamente después de ingresar control-C. La salida para el test 2 debe ser:

```
Test 2: Un ejemplo grande con 33 ciudades
Calculando secuencialmente
^CResultado final: nperm=100000000 distancia
minima=11.058070
Ruta: 0 26 4 3 25 28 2 9 12 17 31 1 21 8 14 19 29 23 7 22
10 27 16 11 6 18 30 32 24 13 15 20 5 0
Resultado insuficiente, debio ser menor que 10.800000
```

Pruebe nuevamente recompilando su tarea con las opciones de optimización con: *make test-O*. En esta tarea no necesita entregar el archivo *resultados.txt*, probar en 32 bits o probar con valgrind.

### Entrega

Ud. debe entregar los archivos *t8a.c* y *t8b.c* comprimidos en el archivo *t8ab.zip*. No incluya otros archivos por favor. Su tarea debe compilar y ejecutar sin errores o warnings en *Debian 10 de 64 bits*. No se aceptarán tareas que no funcionen. Se descontará medio punto por día de atraso. No se consideran los días sábado, domingo o festivos.