

1. Problema, objetivos y restricciones

El problema a resolver será el Bus Evacuation Problem definido en [1], donde se considera que los puntos de encuentro y refugios están definidos al momento de planificar la evacuación, y el número de evacuados es conocido. La idea principal es que luego de implementar su algoritmo puedan comparar sus resultados con los de la literatura.

1.1. Objetivo del problema

El objetivo del problema es planificar la ruta de los buses minimizando el tiempo total de evacuación. Dicho de otro modo, se debe minimizar el tiempo de evacuación del bus que más se demora en hacer todos sus recorridos.

1.2. Restricciones

Se deben respetar los límites de capacidad de buses y refugios. Además, todos los individuos deben ser llevados de los puntos de encuentro a los refugios.

2. Instancias de prueba

Las instancias de prueba que se utilizarán han sido generadas en orden creciente de dificultad, con valores aleatorios. Varias de las instancias están basadas en [1] para que puedan comparar sus resultados. En las próximas semanas se les proporcionará una instancia adicional basada en un escenario real. Su implementación debe resolver **como mínimo** las tres primeras instancias. En la siguiente tabla se muestran los datos de cada instancia, donde E es el número de estaciones, P el número de Puntos de encuentro, R el número de refugios y B el número de buses disponibles.

N° Instancia	E	P	R	B	Fuente
1	1	4	2	4	[1]
2	1	5	3	6	[1]
3	2	9	7	5	Ninguna
4	2	12	3	6	[1]
5	2	22	4	10	[1]
6	2	32	5	18	[1]
7	3	11	10	7	Ninguna
8	5	25	12	15	Ninguna
9	8	40	20	20	Ninguna

Figura 1: Datos de instancias generadas.

2.1. Formato de instancias de prueba

El formato de un archivo de instancia es como se muestra a continuación:

[B]: [capacidadBuses]
 [E]: [#busesEstación1] [#busesEstación2] ... [#busesEstaciónE]
 [P]: [#personasTotalPE]: [#personasPE1] [#personasPE2] ... [#personasPEP]
 [R]: [capacidadTotalR]: [capacidadRefugio1] [capacidadRefugio1] ... [capacidadRefugioR]

 [Estación1]: [dist_Estación1-PtoEncuentro1] [dist_Est1-PtoEnc2] ... [dist_Est1-PtoEncP]
 [Estación2]: [dist_Estación2-PtoEncuentro1] [dist_Est2-PtoEnc2] ... [dist_Est2-PtoEncP]
 (...)
 [EstaciónE]: [dist_EstaciónE-PtoEncuentro1] [dist_EstE-PtoEnc2] ... [dist_EstE-PtoEncP]

 [PtoEncuentro1]: [dist_PtoEncuentro1-Refugio1] [dist_PtoEnc1-Ref2] ... [dist_PtoEnc1-RefR]
 [PtoEncuentro2]: [dist_PtoEncuentro2-Refugio1] [dist_PtoEnc2-Ref2] ... [dist_PtoEnc2-RefR]
 (...)
 [PtoEncuentroP]: [dist_PtoEncuentroP-Refugio1] [dist_PtoEncP-Ref2] ... [dist_PtoEncP-RefR]
 (el archivo termina con un salto de línea que delimita el fin del archivo)

En el primer bloque se muestran las cantidades de buses B, de estaciones E, de puntos de encuentro P y de refugios R utilizados, seguidos de las capacidades (para buses y refugios) o “nivel de ocupación” (para puntos de encuentro y estaciones) de cada uno. Para los puntos de encuentro y refugios, además se menciona el número total de personas a evacuar (suma de las demandas en cada PE) y la capacidad total de los refugios (suma de las capacidades de cada refugio) respectivamente.

En el segundo bloque se muestra la matriz de distancias entre cada estación (fila) y cada punto de encuentro (columnas). Para cada fila se comienza indicando el número de la estación, seguido de dos puntos, y las distancias de esa estación a cada punto de encuentro.

El tercer y último bloque contiene la matriz de distancias de puntos de encuentro (filas) a refugios (columnas). En cada fila se indica inicialmente el número del punto de encuentro, seguido de dos puntos, y la distancia de este a cada refugio.

Por ejemplo, el archivo de instancia “*InstanceBEP-3-11-10-7.txt*” es el siguiente:

```

7: 36
3: 1 5 1
11: 684: 72 36 36 108 108 72 36 72 72 36 36
10: 756: 36 72 144 72 72 72 144 72 36 36

1: 8 7 9 2 6 10 10 8 10 10 4
2: 10 1 3 4 8 3 5 2 1 4 1
3: 8 4 6 8 3 9 3 6 9 9 3

1: 6 10 7 7 5 7 6 1 1 7
2: 2 7 4 6 2 2 8 9 1 2
3: 7 10 1 7 6 1 1 9 6 7
4: 6 5 4 8 7 1 5 4 7 6
(...)
11: 2 7 3 2 2 10 10 8 8 1

```

Se puede observar que en esta instancia se consideran 7 buses de capacidad 36 cada uno; 3 estaciones, cada una con 1, 5 y 1 bus respectivamente; 11 puntos de encuentro, con un total de 684 personas a evacuar; y 10 refugios, con una capacidad total de 756 posibles albergados. En el primer punto de encuentro hay 72 personas, en el segundo hay 36, en el tercero 36, y así sucesivamente. El primer refugio tiene una capacidad para 36 personas, el segundo para 72, el tercero para 144, etc.

En el segundo bloque, las distancias de la estación 1 a cada punto de encuentro son: 8 al punto de encuentro 1, 7 al punto de encuentro 2, 9 al punto de encuentro 3, etc. Igualmente con las distancias de las siguientes estaciones a cada punto de encuentro.

Finalmente, en el tercer bloque, las distancias del punto de encuentro 1 a cada refugio son: 6 al refugio 1, 10 al refugio 2, 7 al refugio 3, etc. Así mismo con las distancias de los siguientes puntos de encuentro a cada refugio.

2.2. Características de instancias de prueba

Las instancias de prueba tienen las siguientes características:

1. El número de evacuados en cada punto de encuentro es múltiplo de la capacidad de los buses [2].
2. Las capacidades de los refugios son múltiplos de la capacidad de los buses [2].
3. Los puntos de encuentro no están conectados entre sí.
4. Los refugios no están conectados entre sí.
5. Las distancias de un nodo a otro son las mismas de ida y de vuelta. Es decir, la distancia del punto de encuentro i al refugio j es igual a la distancia del refugio j al punto de encuentro i .
6. La capacidad total de los refugios siempre es mayor o igual a la demanda total en los puntos de encuentro.
7. La cantidad de buses en cada estación es variable.
8. En las instancias basadas en [1] se usa la misma capacidad para los buses, de 20 personas cada uno. La demandas de los puntos de encuentro, capacidades de los refugios y cantidad de buses en cada estación son, sin embargo, aleatorias, ya que en el paper no se mencionan sus valores.

3. Especificaciones del input del programa

El programa debe recibir como input (en línea de comandos) los siguientes parámetros:

1. La ruta del archivo de instancia, por ejemplo :“instancias/InstanceBEP-1-4-2-4.txt”

Usted puede añadir otros parámetros adicionales en la entrada, tal como cantidad de iteraciones máxima, tiempo máximo de ejecución, temperatura, cantidad de restarts, etc. Si lo hace, **debe incorporarlos junto con una descripción en el archivo README.txt**, estableciendo al menos un ejemplo de cómo ejecutar su programa con estos parámetros.

4. Especificaciones del output del programa

Su programa debe mostrar por pantalla:

1. El valor de la función objetivo (duración total de la evacuación).
2. Tiempo de ejecución en segundos.
3. Cantidad de iteraciones alcanzada (si aplica).

Además de lo anterior, se pide que genere un archivo de texto con los datos anteriores, además de lo siguiente:

1. Una “tabla” similar a la mostrada en el ejemplo 1 en [2], donde se detalle la ruta completa seguida por cada bus, añadiendo una columna donde se muestren los tiempos de evacuación de cada bus. Debe marcar el bus que determina la duración de la evacuación (si son varios, márquelos también).
2. El número de personas en cada refugio al final de la ejecución.

Si desea mostrar otros datos que considere relevantes (y que puedan visualizarse fácilmente) puede añadirlos al final del output, **especificando en el archivo README.txt su significado y qué lo motivó a añadirlos a la salida del programa.**

5. Recomendaciones y recordatorios

1. Recuerde que su programa debe ser realizado en C/C++ con un Makefile para poder compilarlo.
2. Recuerde que debe COMENTAR su código.
3. Recuerde incluir un README donde se especifiquen:
 - a) Instrucciones de ejecución.
 - b) Parámetros necesarios para ejecutar el algoritmo, con una explicación de cada uno (significado, objetivo del parámetro, etc.).
 - c) Ejemplo(s) de comando(s) para ejecutar el algoritmo.
 - d) En caso de requerirlo, el detalle de datos adicionales incluidos en el output.
 - e) Cualquier supuesto que considere relevante.
4. Para realizar mediciones de tiempo de su programa hágalo usando “usr/bin/time¹”, donde el tiempo de ejecución total de su programa será “user + sys”. Se recomienda usar un archivo bash para automatizar la ejecución de su programa con distintos parámetros usando time y guardar los resultados en un archivo.
5. Si tiene problemas con algunas instancias específicas producto de, por ejemplo, cálculos muy grandes o demasiada memoria ram utilizada que saturó su pc, se debería hacer lo siguiente:
 - a) Explicar en el informe cuáles instancias fueron las que originaron el problema

¹Para más información revisar la documentación oficial

- b) Explicar cuál fue el problema que se detectó: tamaño de la instancia muy grande, error aleatorio, etc.

Referencias

- [1] Douglas R Bish. Planning for a bus-based evacuation. *OR spectrum*, 33(3):629–654, 2011.
- [2] Marc Goerigk, Bob Grün, and Philipp Heßler. Branch and bound algorithms for the bus evacuation problem. *Computers & Operations Research*, 40(12):3010–3020, 2013.