

Universidad Técnica Federico Santa María  
Departamento de Informática

# Instrucciones Entregable #2 SMP

Inteligencia Artificial

*Javiera Loyola*  
*javiera.loyola@alumnos.usm.cl*  
Junio 2017

# 1 Problema, objetivos y restricciones

El problema a resolver será SMTI, es decir Stable Marriage Problem with Incomplete Lists and Ties. La idea principal es que luego de implementar su algoritmo puedan comparar sus resultados con los de la literatura. Para esto deberán resolver Max SMTI: encontrar el emparejamiento débilmente estable de máxima cardinalidad (este problema es NP-Hard)[4, 2, 3, 1].

## 1.1 Objetivo del problema

El objetivo del problema es maximizar la cantidad de parejas formadas.

## 1.2 Restricciones

El emparejamiento final debe ser débilmente estable.

# 2 Instancias de prueba

Las instancias de prueba que se utilizarán serán un subconjunto de las provistas por el sitio [http://is.muni.cz/th/172646/fi\\_m?furl=%2Fth%2F172646%2Ffi\\_m;so=nx;lang=en](http://is.muni.cz/th/172646/fi_m?furl=%2Fth%2F172646%2Ffi_m;so=nx;lang=en), si tienen problema para descargarla de docencia, pueden encontrarlas en el siguiente enlace: [https://www.dropbox.com/s/xuimm4y82pdakl0/instances\\_smp.zip?dl=0](https://www.dropbox.com/s/xuimm4y82pdakl0/instances_smp.zip?dl=0)

## 2.1 Formato de instancias de prueba

El formato de un archivo de instancia comienza describiendo las listas de preferencias de los hombres, la forma de describir las distintas listas es la siguiente:

```
1: 83 42 79 3 ( 61 85 80 63 50 98 97 14 70 ) 99 36 35 5 ( 77 56 ) ( 30 19 ) ( 73 49 ) 93
2: ( 36 83 ) 47 9 56 80
3: 10 ( 21 73 ) ( 66 91 40 51 ) ( 76 94 3 97 81 39 33 ) 18 99 ( 54 9 100 85 84 93 ) ( 24 29 63 )
4: 73 45 36 17 40 68 ( 32 79 ) 93 89 ( 24 78 75 34 ) 64 61 28 ( 29 84 56 42 60 ) ( 10 39 33 ) 86 87 80 99
9 66 77 71 ( 25 5 ) ( 41 2 51 ) 44 3 16
5: ( 34 63 51 26 73 16 46 ) ( 40 28 ) 68 ( 99 79 82 60 87 5 86 24 94 17 44 ) 42 ( 89 29 32 ) ( 9 2 45 ) (
47 39 )
6: 31 40 ( 9 83 93 44 82 88 87 ) 89 51
7: 64
(...)
100: 98 71 75 24 ( 99 32 20 ) 44 48 ( 90 28 ) ( 88 38 66 93 ) ( 97 87 )
(salto de línea que delimita el fin del grupo de los hombres y el comienzo del grupo de las mujeres)
1: ( 86 69 ) 99 ( 65 94 90 50 ) ( 34 13 14 ) 20 63
2: 41 47 72 60 ( 59 85 ) ( 86 49 28 4 30 ) ( 37 44 ) ( 16 14 ) ( 5 73 ) 67 43 62 89 48
3: ( 81 73 ) 13 99 ( 82 28 95 57 85 63 ) ( 59 89 76 96 47 ) 3 ( 29 14 ) 15 64 1 4
4: 57 88 59 10 ( 45 24 ) 29 75
(...)
100: 47 ( 22 46 ) 10 84 93 44 59 ( 3 60 ) ( 73 18 ) 25 ( 81 67 ) 79 42 ( 8 29 50 49 72 19 ) 57 98 ( 24 33
54 )
```

(el archivo termina con un salto de línea que delimita el fin del archivo).

## 2.2 Características de instancias de prueba

Las instancias de prueba tienen las siguientes características:

1. La mayor parte de las instancias son de tamaño 100 (100 hombres y 100 mujeres). Excepto las últimas cuyo tamaño es superior o igual 1000.
2. Algunas de las listas de preferencias pueden estar incompletas.
3. Algunas de las listas de preferencias pueden tener ties: estos están especificados entre paréntesis.
4. Algunas de las instancias, especialmente las últimas, pueden tener hombres o mujeres cuya lista de preferencia está vacía (i.e no prefieren a nadie del otro grupo). En cuyo caso, sólo se indica en la línea el número de la persona seguido de dos puntos, por ejemplo “24:”.
5. Existen diferentes conjuntos de instancias que están separados por carpetas, con distintas características:
  - (a) Chart1 : instancias de SMTI random. El tamaño de las instancias es 100.
  - (b) Chart4 : instancias de SMTI creadas mediante el algoritmo de Gale & Shapley. La característica principal de estas instancias es que es seguro que pueden tener un match estable de tamaño 100 (Instancias óptimas). El tamaño de las instancias es 100.
  - (c) Chart16 : instancias de SMTI random. El tamaño de las instancias es 1000.
  - (d) Chart17 : instancias de SMTI random. El tamaño de las instancias es 2000.
  - (e) Chart18 : instancias de SMTI random. El tamaño de las instancias es 3000.

## 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 :“instances/Chart1/instance\_3.txt”
2. La máxima cantidad de iteraciones permitida en una ejecución de su programa (no aplica para técnicas completas).

Ejemplo de entrada: “./SMTI instances/Chart1/instance\_3.txt 1000”, donde SMTI es el nombre de su programa , “instances/Chart1/instance\_3.txt 1000” la ruta completa de la instancia a resolver y 1000 la cantidad de iteraciones.

Es preciso mencionar que se pueden añadir otros parámetros adicionales en la entrada, 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 (cantidad de parejas de su solución).
2. Cantidad de iteraciones alcanzada.
3. Si se ha alcanzado un óptimo: si el óptimo ha sido alcanzado mostrar “optimo”, sino mostrar “no\_optimo”.
4. Si su mejor solución encontrada es estable: si su mejor solución es estable mostrar “estable”, sino mostrar “no\_estable”.
5. Adicionalmente, si usted desea mostrar otros datos que sean importantes (y que puedan visualizarse fácilmente) puede mostrarlos al final del output, **especificando en el archivo README.txt qué quieren decir y qué sentido tiene añadirlos a la salida del programa.**

## 5 Recordatorios y recomendaciones

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. Para realizar mediciones de tiempo de su programa hágalo usando “usr/bin/time<sup>1</sup>”, 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.
4. Si tiene problemas con algunas instancias específicas producto de, por ejemplo, cómputos muy grandes o demasiada memoria ram utilizada que saturó su pc, usted debería hacer lo siguiente:
  - (a) Explicar en el informe cuáles instancias fueron las que originaron el problema
  - (b) Explicar cuál fue el problema que usted detectó: tamaño de la instancia muy grande, error aleatorio, etc.
5. Se recomienda leer la sección de experimentos de [4], en donde comparan diversos algoritmos de aproximación tomando como base las mismas instancias con las que usted trabajará.
6. **Recuerde que el segundo entregable consta de tres cosas: El CD con el código y el informe realizado en digital, el informe impreso y el informe 1 con las correcciones solicitadas (este último para poder revisar si usted hizo los cambios solicitados).**

---

<sup>1</sup>Para más información revisar la documentación oficial

## References

- [1] Mirco Gelain, Maria Silvia Pini, Francesca Rossi, Kristen Brent Venable, and Toby Walsh. Local search for stable marriage problems with ties and incomplete lists. In *PRICAI 2010: Trends in Artificial Intelligence*, pages 64–75. Springer, 2010.
- [2] Ian P Gent and Patrick Prosser. An empirical study of the stable marriage problem with ties and incomplete lists. In *ECAI*, pages 141–145, 2002.
- [3] Ian P Gent, Patrick Prosser, Barbara Smith, and Toby Walsh. Sat encodings of the stable marriage problem with ties and incomplete lists. *SAT*, 2002:133–140, 2002.
- [4] Andrej PODRRADSKY. Stable marriage problem algorithms. Diplomova prace, Masarykova univerzita, Fakulta informatiky, 2011 [cit. 2014-06-12].