

# Obligatorio 4

## Programación 3

### Curso 2011

#### Versión 1.1

## 1 Objetivos

- Diseñar e implementar un algoritmo basado en la técnica de Backtracking.

## 2 Conocimientos previos

- Metodología de programación estructurada
- Conocimientos básicos de C\*
- Backtracking

## 3 Descripción del problema

"Busco Amor" es una empresa que se dedica a formar parejas entre clientes solitarios e intenciones serias. Esta pequeña empresa es manejada por su creadora Doña Emma, una casamentera de corazón, quien decide con su instinto cupidezco que clientes unir. Doña Emma ha decidido que es hora de que la empresa crezca y se modernice. Para esto solicita nuestra ayuda, ya que la cantidad de clientes que la empresa espera manejar escapa de las manos de la dueña. La empresa esta necesitando contar con un sistema que utilizando ciertas reglas y atributos descriptivos de los clientes, genere distintas alternativas de parejas que luego serán chequeadas por Doña Emma. La idea para este laboratorio entonces es hacer participe a los estudiantes en la ardua y no siempre grata tarea del casamentero.

### 3.1 Modelo

Los clientes son divididos en dos conjuntos C1 y C2, de tamaños |C1| y |C2| respectivamente. El objetivo es hacer la máxima cantidad de parejas posibles entre clientes de distintos conjuntos optimizando según uno de los criterios descritos más abajo.

De cada cliente se conocen ciertos atributos propios (P) y ciertos atributos de la persona que creen que buscan (B). El modelo maneja tres atributos: Apariencia (A), Inteligencia (I) y Simpatía (S). Se puntúa cada atributo con un valor en el rango de 1 al 10, siendo 1 el mínimo y 10 el máximo. El modelo además utiliza dos índices, el índice (D) y el (Q). El índice D mide para un cliente, la diferencia entre la pareja esperada y la pareja obtenida. El índice Q es la suma de los índices D de cada cliente de la pareja. Si tenemos entonces la pareja (c1, c2), los índices D y Q se calculan de la siguiente manera,

$$Q = D1 + D2$$

$$D1 = |c1.B.A - c2.P.A| + |c1.B.I - c2.P.I| + |c1.B.S - c2.P.S|$$

$$D2 = |c2.B.A - c1.P.A| + |c2.B.I - c1.P.I| + |c2.B.S - c1.P.S|$$

Por supuesto existen ciertas restricciones al momento de unir dos clientes. Si bien no hay edad para el amor, en la práctica Doña Emma maneja límites para la diferencia de edad de una pareja. Definimos entonces cuatro límites (Lm1), (LM1), (Lm2) y (LM2), que establecen la diferencia de edad máxima entre una pareja. Si (E) referencia la edad de los clientes entonces toda pareja (c1, c2) debe cumplir,

$$\begin{aligned}c1.E - [Lm1] &< c2.E < c1.E + [LM1] \\c2.E - [Lm2] &< c1.E < c2.E + [LM2]\end{aligned}$$

Por último, Doña Emma es muy cabalera y siempre ha unido parejas respetando al menos uno de los deseos de los clientes en cuanto a color de ojos o de cabello y desea que esta tradición se mantenga. Por lo tanto, de cada cliente se tienen datos sobre su color de cabello (H) y ojos (O) y los de la pareja esperada. El modelo maneja cuatro colores de cabello Castaño, Rubio, Negro y Rojo, y cinco colores de ojos Marrón, Verde, Azul, Negro y Gris.

### 3.2 Criterio clásico

El criterio clásico construye parejas de manera de que globalmente los clientes satisfagan sus expectativas. El objetivo de este criterio es entonces minimizar la suma de los Q de las parejas que se forman.

### 3.3 Criterio de los opuestos

Este es uno de los secretos mas guardados por Doña Emma y es el criterio que mas parejas felices ha formado. Doña Emma cree que las personas suelen tener un modelo establecido inconscientemente al momento de buscar su media naranja. El criterio de los opuestos según ella intenta formar parejas de manera de romper estos modelos que obviamente han fallado para estos clientes. La idea del modelo de los opuestos es maximizar la suma de los Q de las parejas que se forman.

### 3.4 Flexibilizando cábalas

Hemos convencido a Doña Emma de flexibilizar el asunto de la cábala para así intentar formar más parejas. La idea es penalizar en ambos criterios a las parejas que no cumplan la cábala en vez de no tenerlas en cuenta. Utilizaremos el valor de penalización (M).

## 4 Se pide

Formalizar y resolver los problemas planteados aplicando la técnica de Backtracking. Implementar las funciones especificadas en el archivo buscoamor.h.

## 5 Lenguaje a utilizar

El lenguaje a utilizar en este trabajo será C con las siguientes extensiones:

- Operadores `new` y `delete`.
- Pasaje de parámetros por referencia (uso de `&`).
- Declaración de tipos como en C++ para registros y enumerados.
- Sobrecarga de funciones.
- Uso de `cin` y `cout`.
- Uso del tipo `bool` predefinido en C++.

## 6 Qué se espera

Para cada módulo de cabecera (`.h`) con los prototipos de las operaciones solicitadas, debe entregarse un módulo (`.cpp`) con la implementación de dichas operaciones. Debe respetarse estrictamente los prototipos especificados, esto es: nombre de la operación, tipo, orden y forma de pasaje de los parámetros y tipo de retorno. Los **módulos de cabecera** pueden bajarse de la página web del curso ([www.fing.edu.uy/inco/cursos/prog3](http://www.fing.edu.uy/inco/cursos/prog3)). Estos módulos no forman parte de la entrega, y por lo tanto, **no deben ser modificados**.

**Los módulos deben funcionar en el ambiente MinGW instalado en facultad.** Se espera que todos los módulos compilen sin errores (utilizando las flags “-Wall” y “-Werror”), se ejecuten sin colgarse y den los resultados correctos.

## 7 Entregas semanales

Las entregas semanales se realizarán en los respectivos monitoreos.

- **Primer semana**

Formalizar los problemas en términos de Backtracking. Indicar: forma de la solución, restricciones explícitas e implícitas, función objetivo y predicados de poda.

## 8 Forma de la entrega final

Se deberá entregar únicamente los archivos (respetando las mayúsculas en los nombres):

- `buscoamor.cpp`

No se podrá entregar otros archivos que no sean estos.

La primera línea de cada uno de los archivos debe contener un comentario (`/* ... */`) con la cédula de los autores, sin puntos ni dígito de verificación. Por ejemplo, si las cédulas son 1.234.567-8, 4.254.566-2 y 3.339.717-0, la primera línea de cada archivo deberá ser exactamente:

```
/* 1234567 4254566 3339717 */
```

## 9 Advertencia sobre el manejo de la memoria

Cuando un programa contiene errores en el manejo de la memoria, su comportamiento puede ser inestable. Esto implica que algunas veces funciona correctamente y otras no. En ciertos casos esto puede inducir a creer (erróneamente) que ciertos programas, que en realidad son incorrectos, funcionan correctamente. Este aspecto es influenciado, entre otras cosas, por el *sistema operativo* en el que se ejecuten los programas. Recomendamos tener sumo cuidado con este punto y testear los módulos en sistemas operativos Windows NT, Windows 2000 o Windows XP. CON LA VERSION DE MINGW INSTALADA EN LAS SALAS DE INFORMÁTICA DE LA FACULTAD.

## 10 Sobre la individualidad del trabajo

Para este obligatorio rige el reglamento de No Individualidad publicado en la página web del curso.

## 11 Fecha de entrega

El trabajo debe entregarse el día **domingo 30 de octubre del 2011** antes de las **22:00** horas. La entrega se realizará mediante un formulario que se habilitará oportunamente en la página web del curso.