

# Resumen de algoritmos para maratones de programación

Diego Alejandro Martínez - Manuel Felipe Pineda

1 de octubre de 2012

## Índice

### 1. Plantilla

### 2. Grafos

- 2.1. Dijkstra . . . . .
- 2.2. Bellman-Ford . . . . .
- 2.3. Floyd-Warshall . . . . .
- 2.4. Johnson . . . . .
- 2.5. Minimum Spanning Tree: Kruskal . . . . .
- 2.6. Minimum Spanning Tree: Prim . . . . .
- 2.7. Breadth First Search . . . . .
- 2.8. Depth First Search . . . . .
- 2.9. Strongly Connected Components . . . . .
- 2.10. Puntos de articulación . . . . .
- 2.11. 2-SAT . . . . .
- 2.12. Maximum bipartite matching . . . . .
- 2.13. Flujo Máximo . . . . .
- 2.14. Lowest Common Ancestor: TarjanOLCA . . . . .

### 3. Matemáticas

### 4. Geometría

### 5. Strings

### 6. Teoría de Juegos

### 7. Estructuras de Datos

### 8. Hashing

### 9. Misceláneo

## 1. Plantilla

```
1  #include <cstdio>
1  #include <cmath>
2  #include <iostream>
2  #include <string>
2  #include <vector>
2  #include <queue>
2  #include <stack>
2  #include <list>
2  #include <map>

2  using namespace std;

2  #define all(x) x.begin(),x.end()
2  #define rep(i,a,b) for(int i=a;i<b;i++)
2  #define REP(i,n) rep(i,0,n)
2  #define foreach(x, v) for (typeof (v).begin() x = (v).begin(); \
2  x != (v).end(); ++x)
2  #define D(x) cout << #x " = " << x << endl;

2  typedef long long int lld;
2  typedef pair<int,int> pii;
2  typedef vector<int> vi;
2  typedef vector<pii> vpii;

2  int main(){
2
2      return 0;
2  }
2  .....
```

## **2. Grafos**

2.1. Dijkstra

2.2. Bellman-Ford

2.3. Floyd-Warshall

2.4. Johnson

2.5. Minimum Spanning Tree: Kruskal

2.6. Minimum Spanning Tree: Prim

2.7. Breadth First Search

2.8. Depth First Search

2.9. Strongly Connected Components

2.10. Puntos de articulación

2.11. 2-SAT

2.12. Maximum bipartite matching

2.13. Flujo Máximo

2.14. Lowest Common Ancestor: TarjanOLCA

## **3. Matemáticas**

## **4. Geometría**

## **5. Strings**

## **6. Teoría de Juegos**

## **7. Estructuras de Datos**

## **8. Hashing**

## **9. Miseláneo**