

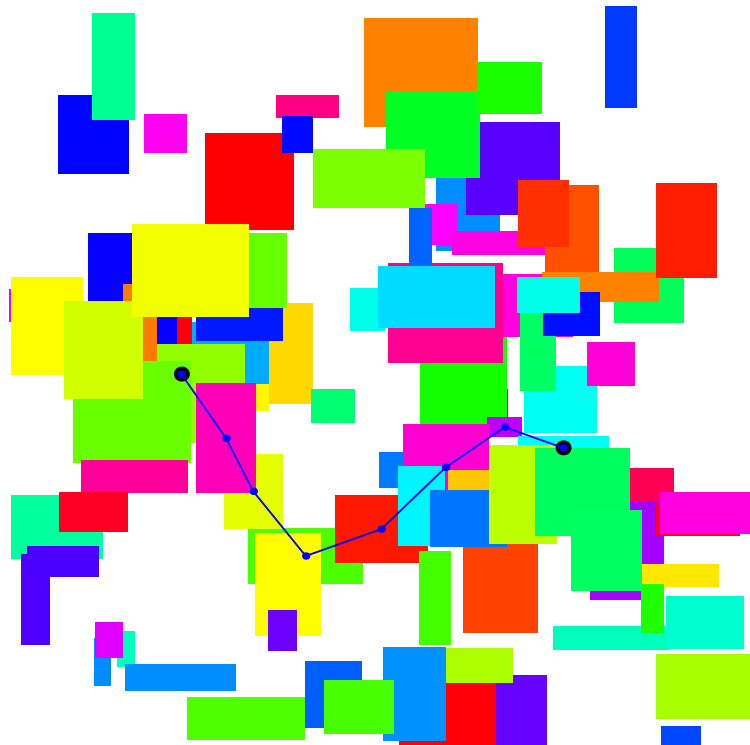
Corso Tecniche di Programmazione Avanzata

Esame del 8 luglio 2019

Esercizio B

Problema

Nonna Adelina ha piastrellato la cucina, essendo molto vecchia non si è accorta che ha usato piastrelle di varia misura e disposto le piastrelle in qualche maniera. A volte le piastrelle si sovrappongono e ci sono molti buchi (vedi figura):



supponiamo ora di numerare le piastrelle da 1 a n , diremo che la piastrella i è collegata alla piastrella j se le piastrelle si sovrappongono. Il problema proposto è quindi:

Data una piastrellazione come descritto in precedenza e due punti $(x_0, y_0), (x_1, y_1)$ dire se i punti sono su una o più piastrelle e collegati da una successione di piastrelle sovrapposte.

Una piastrella è definita dalla struttura

```
typedef struct {  
    double xmin, xmax;  
    double ymin, ymax;  
} piastrella;
```

e la funzione che risolve il problema ha prototipo

```
#include <vector>

bool
connected(
    std::vector<piastrella> const & board,
    double x0, double y0,
    double x1, double y1,
    std::vector<unsigned> const & path
);
```

Suggerimento

Controllare che i punti cadano all'interno di una o più piastrelle. In caso positivo costruire il grafo delle connessioni delle piastrelle e tramite spanning tree vedere se i punti sono connessi.

Per controllare se due piastrelle si sovrappongono potete usare questa funzione:

```
bool
doOverlap( piastrella const & A, piastrella const & B ) {
    // If one rectangle is on left side of other
    if (A.xmin > B.xmax || B.xmin > A.xmax) return false;

    // If one rectangle is above other
    if (A.ymin > B.ymax || B.ymin > A.ymax) return false;

    return true;
}
```

Punteggio

Ci sono 8 test, ogni test da un punteggio 1.0 se il problema è risolto 0 altrimenti.