In questa sezione descriviamo i test unitari che abbiamo implementato.

Abbiamo deciso di eseguire in tutto 9 test suddivisi per “categoria”: tre test sui costruttori, due test sugli algoritmi di sorting, due test sull’import e un test sull’algoritmo di raffinamento.

Test sui costruttori (TestConstructors):

1. TestVertex: verifica il corretto funzionamento del costruttore dei vertici,
2. TestEdge: verifica il corretto funzionamento del costruttore dei lati;
3. TestTriangle: verifica il corretto funzionamento del costruttore dei triangoli.

Test sugli algoritmi di sorting (TestSorting):

Sono stati eseguiti su vettori di int (vector<int>):

1. TestHeapSort: verifica la corretta implementazione dell’algoritmo dell’Heapsort;
2. TestInsertion: verifica la corretta implementazione dell’algoritmo dell’Insertion Sort.

Test sull’import (TestImport):

Abbiamo creato tre file fittizi contenenti vertici, lati e triangoli, inventati per i test:

* 0dtry.csv, contenente i vertici e avente la stessa struttura del file Cell0Ds.csv;
* 1dtry.csv, contenente i lati e avente la stessa struttura del file Cell1Ds.csv;
* 2dtry.csv, contenente i triangoli e avente la stessa struttura del file Cell2Ds.csv;

I test eseguiti sono i seguenti:

1. TestVertices: verifica la corretta importazione dei vertici;
2. TestEdges: verifica la corretta importazione dei lati;
3. TestTriangles: verifica la corretta importazione dei triangoli;

Test sull’algortimo di raffinamento (TestRefine):

1. TestBisect: verifica il corretto funzionamento della funzione Bisect.

Nonostante siano test unitari, abbiamo dovuto utilizzare i costruttori anche nei test sull’import e nel test sull’algoritmo di raffinamento perché le funzioni testate sono state definite per lavorare con oggetti da noi creati e quindi non per oggetti generici.