­­PROBLEMA

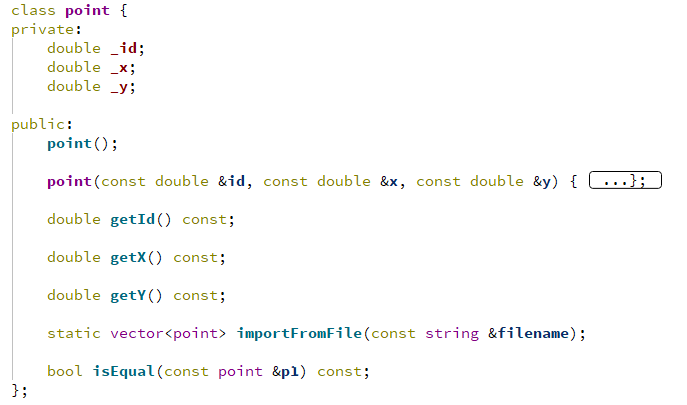
Il problema che ci si pone di fronte è quella di creare a partire da un insieme di punti una sua ricopertura convessa creata da una triangolazione. La triangolazione deve avere caratteristiche tali per cui soddisfa l’ipotesi di Delaunay.

Il processo logico richiede quindi di creare un primo triangolo e a partire da questo completare la triangolazione aggiungendo nuovi triangoli per ogni punto dell’insieme di partenza, verificando per ogni triangolo creato se con il suo adiacente è soddisfatta l’ipotesi di Delaunay. Se tale ipotesi non è rispettata allora si agisce con un flip del lato di adiacenza avendo così la certezza di risolverla.

L’ipotesi di Delaunay chiede che dati due triangoli adiacenti, la somma degli angoli opposti al lato di adiacenza sia minore o uguale a 180 gradi.

SOLUZIONE PROPOSTA

Data la struttura del problema, si è lavorato con unità centrale i triangoli e i suoi elementi. Di conseguenza sono state create le classi dei punti, dei segmenti e dei triangoli. I punti sono caratterizzati da un id, e dalle coordinate x e y. I segmenti dai punti degli estremi. I triangoli invece possono essere rappresentati tramite i tre punti o i tre lati.

Immagine che contiene testo, schermata, Carattere

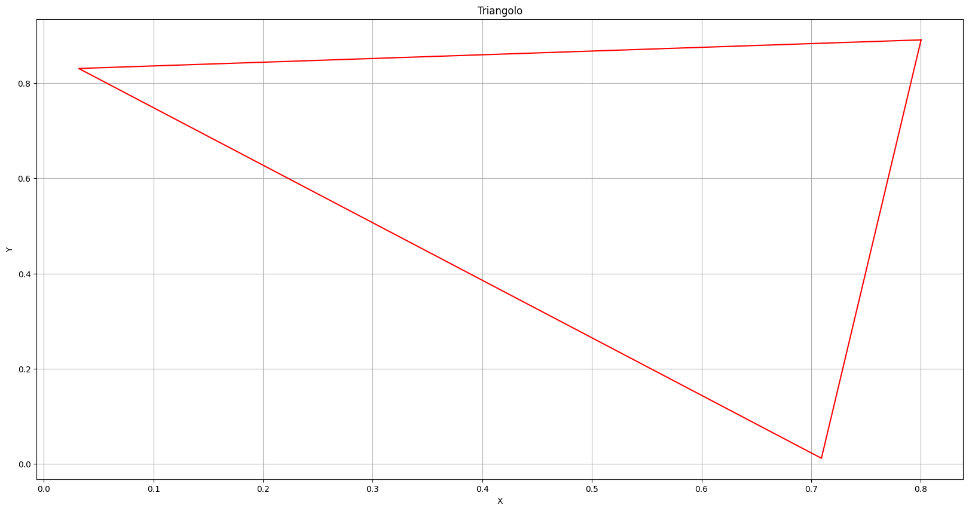
Descrizione generata automaticamente



Per la triangolazione si terrà traccia tramite tre vettori, dei punti, dei segmenti e dei triangoli che verranno aggiunti o tolti con le varie iterazioni.



Prima della fase iterativa dobbiamo andare a cercare il triangolo da cui partire. Questo sarà il triangolo con dimensione massima. Andremo quindi a creare tutti i triangoli possibili dai punti dati per poi calcolarne le aree. Si confrontano una per volta queste aree fino a trovare quella con dimensione maggiore. Sarà questo il primo triangolo della triangolazione.



Il fulcro centrale della soluzione è la verifica della posizione del nuovo punto che andiamo a valutare rispetto alla triangolazione. Il punto potrebbe essere all’interno di un triangolo esistente, su un vertice di un triangolo esistente o esterno da tutti i triangoli e di conseguenza esterno alla triangolazione.

Per ogni caso andiamo a valutare cosa accade.

CASO INTERNO

Se il punto è all’interno di un triangolo già esistente allora i nuovi triangoli si formeranno creando i segmenti che congiungono i punti ai tre vertici del triangolo. Si elimina il triangolo iniziale e si aggiungono alla triangolazione i tre nuovi triangoli se rispetto l’ipotesi di Delaunay. In caso contrario si aggiungeranno i triangoli flippati.

CASO ESTERNO

Se il punto è esterno a tutti i triangoli e quindi a tutta la triangolazione in generale allora si creano tutti i segmenti che uniscono il punto in esame con i punti già appartenenti alla triangolazione, si escludono quelli che creano intersezioni con gli altri segmenti già appartenenti alla triangolazione e con i rimanenti si creano i triangoli ammissibili. Si verifica che valga Delaunay tra il triangolo in questione e i suoi adiacenti. E si aggiungono i nuovi triangoli flippati o no in base alla verifica.

CASO SUL SEGMENTO

Se il punto appartiene ad uno dei segmenti della triangolazione bisogna andare a valutare che genere di segmento è quello preso in esame.

Se il segmento è un segmento di adiacenza e di conseguenza lega due triangoli allora la sottotriangolazione che andiamo a formare sarà costituita da 4 nuovi triangoli. Per ognuno di questi, andremo a valutare Delaunay con i rispettivi triangoli adiacenti e si aggiungono alla triangolazione i nuovi triangoli flippati o meno. Eliminando i triangoli adiacenti iniziali dalla triangolazione e anche il loro segmento di adiacenza.

Se il segmento è esterno rispetto a tutta la triangolazione allora i triangoli creati in questo caso saranno solo 2. Per entrambi anche qui valutiamo Delaunay e aggiungiamo i triangoli flippati o meno. Facendo attenzione a cancellare quello di partenza.

IPOTESI DI DELAUNAY

Per verificare l’ipotesi di Delaunay si prendono due triangoli adiacenti e il loro lato di adiacenza tramite una funzione calcolo gli angoli dei vertici opposti al lato di adiacenza e se la somma di questi è minore o uguale a 180 gradi allora l’ipotesi sarà verificata.

Se l’ipotesi non viene verificata allora utilizzo la funzione di flip che dati due triangoli adiacenti e il loro lato di adiacenza mi restituisce altri due triangoli che saranno quelli con il lato di adiacenza invertito.

Elimino dalla triangolazione il triangolo di partenza e aggiungo i due nuovi forniti dalla funzione flip. Trovo anche il lato di adiacenza tra i due nuovi triangoli e aggiungo questo al vettore che tiene traccia dei segmenti della triangolazione.

Sui nuovi triangoli creati verifico nuovamente che valga l’ipotesi di Delaunay con i loro rispettivi adiacenti.

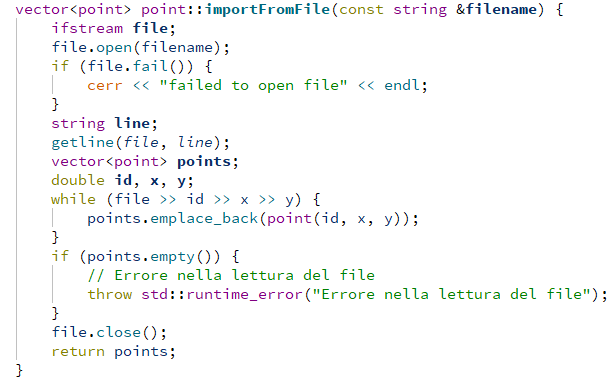
CHIUSURA CODICE

Alla fine, stampo i triangoli ottenuti su un file per poter poi visualizzare la mia triangolazione.

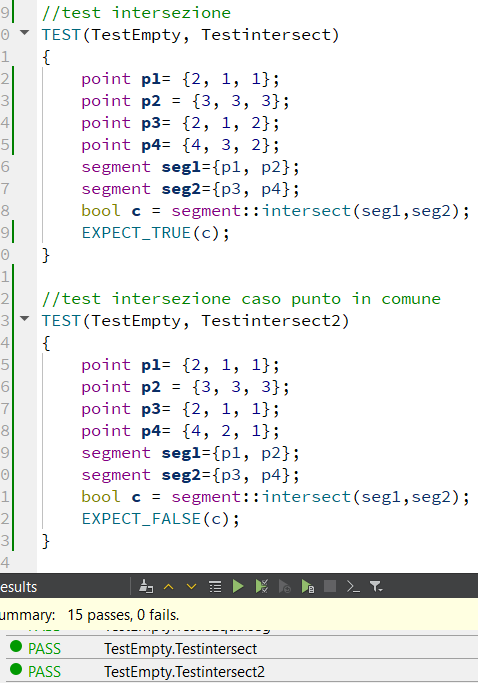
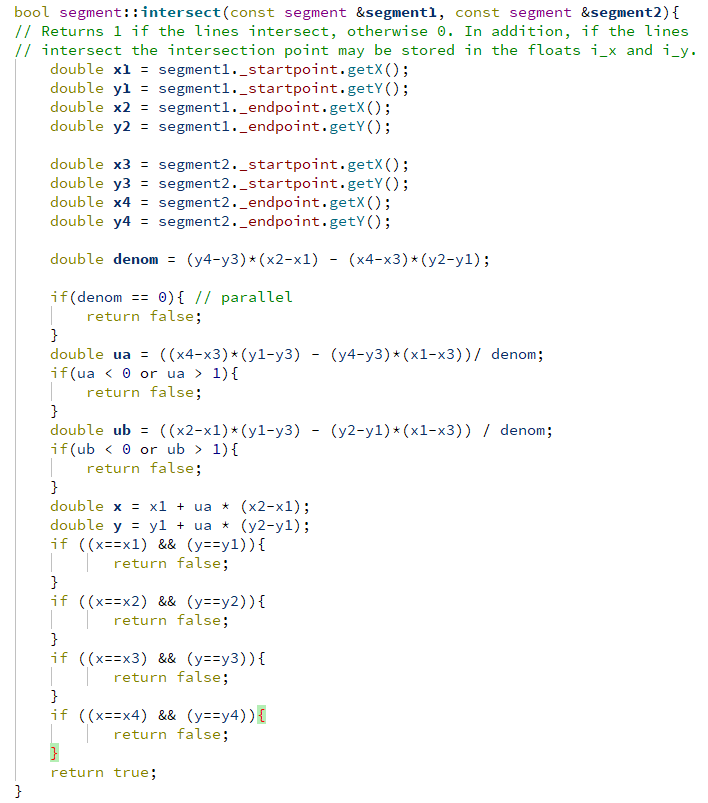
Immagine che contiene linea, diagramma, cerchio, Simmetria

Descrizione generata automaticamente

ALCUNI METODI DEGNI DI NOTA

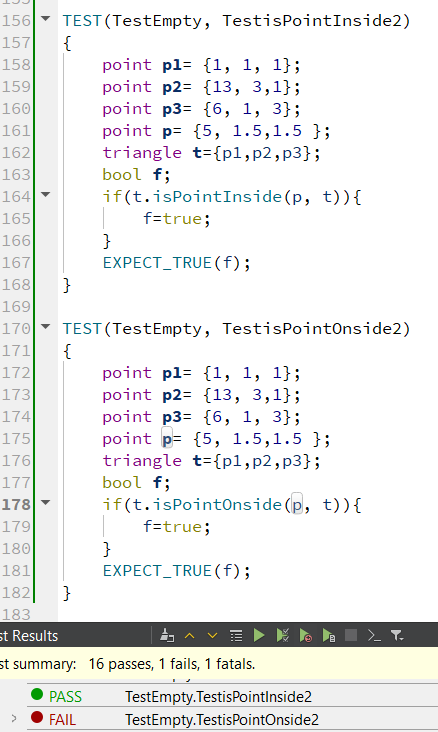
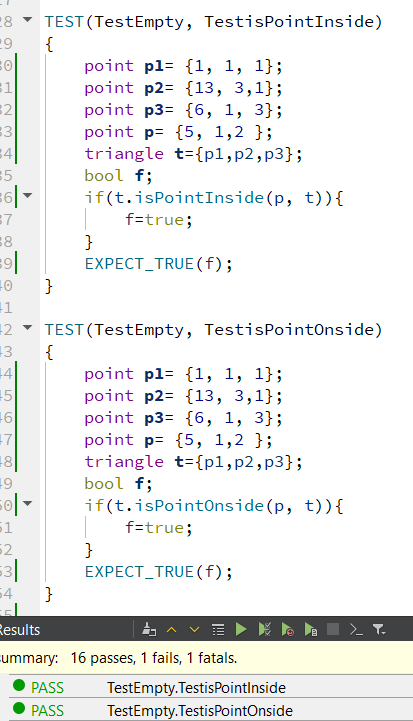


Questo metodo importa da file tutti i punti che ci vengono forniti.



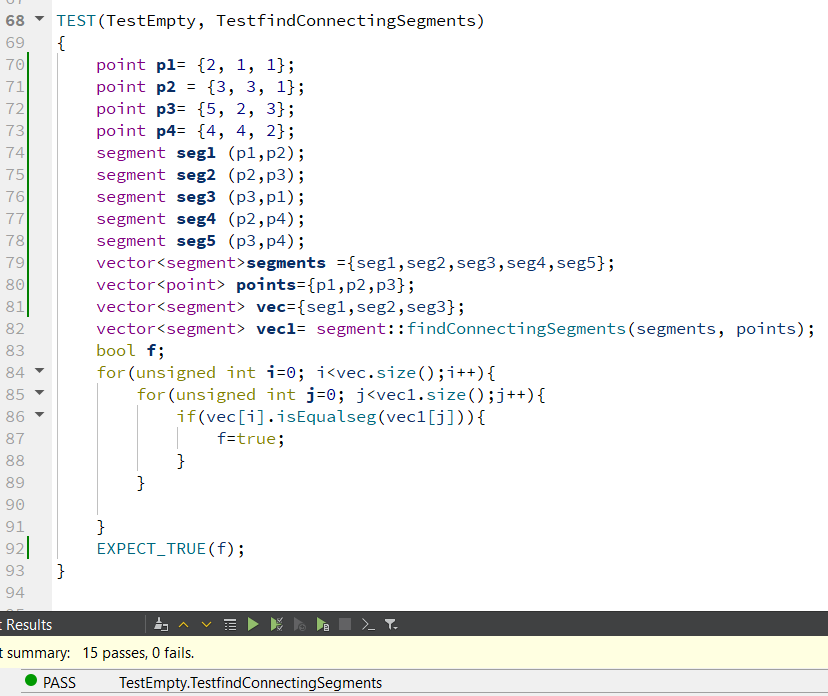
Il metodo intersect è quello che verifica se dati due segmenti tra questi esiste un’intersezione. Eliminando le casistiche in cui i due segmenti hanno uno dei punti di inizio o fine in comune.

Immagine che contiene testo, schermata, numero, Carattere

Descrizione generata automaticamente

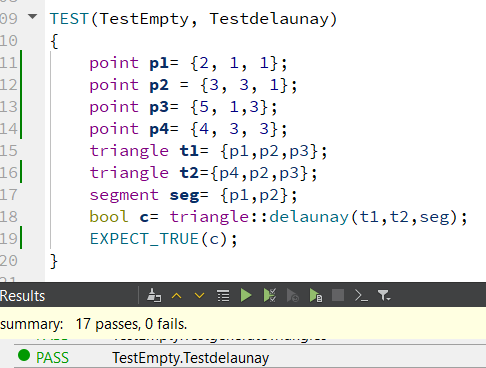
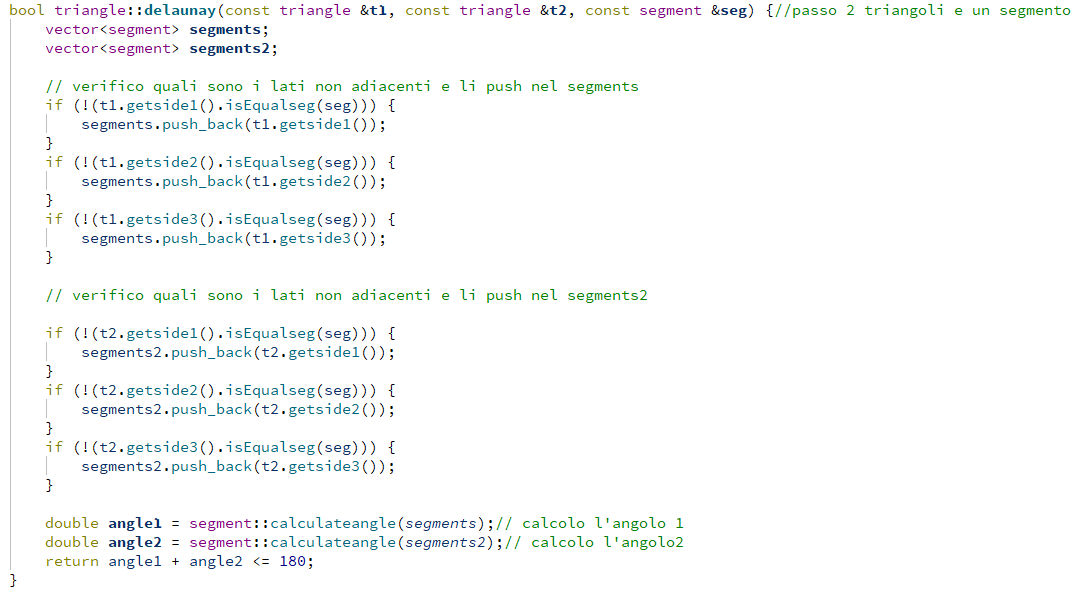
I metodi isPointInside e isPointOnside verificano la posizione del punto rispetto a un triangolo. Il primo ci dice se sta all’interno compreso il caso che sia su un lato, il secondo specifica se è effettivamente su un lato o no.

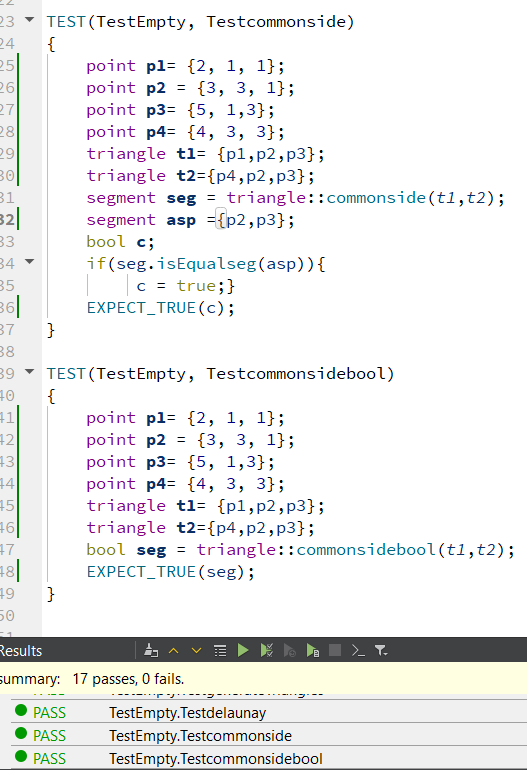
Immagine che contiene testo, schermata, Carattere

Descrizione generata automaticamente

La funzione Delaunay verifica dati due triangoli e il loro segmento di adiacenza se gli angoli opposti al segmento di adiacenza sommano meno di 180 gradi

findConnetingSegments restituisce, dati un insieme di segmenti e un insieme di punti, un vettore contenente i segmenti formati soltanto da coppie di punti dati.





Commonside restituisce il segmento in comune

Commonsidebool ci dice se esiste un segmento in comune



La funzione di flip restituisce, dati due triangoli e il loro segmento di adiacenza, i due triangoli flippati