**1 Documentazione / Commenti**

1.a Generare la documentazione usando Doxygen e verificare se la documentazione prodotta permette di orientarsi nella struttura del progetto software. Indicare i punti che potrebbero essere migliorati.

Il file Doxygen si crea correttamente e sono incluse tutte le classi, ma ci sono due classi isotrapezoid, in cui una rimanda all’altra. Una breve descrizione della classe è presente solo per la classe Polygon.

Classe Polygon: all’interno di alcune funzioni c’è una discordanza tra i parametri specificati nelle definizioni delle funzioni e quelli specificati nei commenti doxygen.

Classe Rectangle: in alcune funzioni mancano i commenti doxygen in cui si specifica che parametro si sta passando.

Class Rhombus: in certe funzioni nel commento dove specificare che cosa ritorna c’è solamente il nome della variabile senza specificare a cosa si riferisce la variabile.

Class isotrapezoid: questa classe non prevede la possibilità di impostare i dati dell’oggetto, come nelle altre classi, prevede però di creare direttamente un oggetto con i dati inseriti, la possibilità di impostare i dati è specificata nel main.

Main: non sono presenti commenti di nessun tipo nel main.

1.b Utilizzando la documentazione e i commenti inseriti nei file .ccp e .h verificare se il progetto software aderisce alle specifiche assegnate durante la prima settimana. Indicare le eventuali specifiche non rispettate.

Generalmente il programma contiene le funzioni minime richieste con anche il controllo che i dati inseriti non creino errori.

Nella classe isotrapezoid viene utilizzata la funzione pow; forse si poteva evitare per alleggerire il programma.

1.c Verificare che il file README introduca correttamente lo scopo del progetto software e che dia sufficienti informazioni per un corretto uso dell’interfaccia a riga di comando. Indicare eventuali mancanze e/o possibili migliorie.

Il file README contiene esclusivamente una riga in cui è specificato il nome della repository su github.

Manca una descrizione totale del funzionamento del programma e qualsiasi tipo di spiegazione dell’utilizzo.

Consiglio l’inserimento di una spiegazione di come gestire l’interfaccia seppur molto facile ed intuitiva.

**2 Compilazione e prima sessione di test**

2.a Verificare se è possibile compilare il progetto

Alla prima compilazione il progetto presenta 39 errori, tutti della stessa tipologia e legati allo switch presente nel main

2.b Test dell’interfaccia a riga di comando: l’interfaccia funziona correttamente? L’interfaccia è di facile utilizzo? Quali prove sono state eseguite per fare il test?

**3 Seconda sessione di test**

3.a Definire una procedura di test che permetta di identificare possibili bug nel codice prodotto per le singole classi e i singoli metodi di ogni classe

3.b Riportare i risultati del test