

## Documento dei requisiti Homework 1

**Committenti:** Matteo Bassani, Alessandro Leonardi

### **Prefazione:**

Il seguente documento è nella sua prima versione, ed è principalmente rivolto ai system engineers, agli end-users ed ai developers. Si è cercato d'utilizzare un linguaggio abbastanza naturale senza fare eccessivamente riferimento a conoscenze specifiche e, laddove è stato fatto, si rimanda al glossario relativo al documento.

### **Introduzione:**

Si vuole realizzare un programma che permetta il posizionamento di tre differenti tipologie di pezzi/pedine su una mappa. Ogni pezzo ha un differente valore di attacco e di difesa in base alla cella in cui è posizionato, ogni cella infatti ha specifiche caratteristiche tipologiche. Il valore di ogni pezzo cambia anche in base alla fascia temporale della mappa.

Le dimensioni della mappa vengono definite in fase di avvio del programma da parte dell'utente.

I dati necessari all'esecuzione vengono prelevati da un file di input, il cui formato verrà specificato in seguito nella sezione requisiti funzionali. I dati vengono letti ed elaborati e vengono svolte delle operazioni su di essi per determinare:

- in quale/i cella/e sono presenti il pezzo o i pezzi con la maggiore difesa di giorno e di notte ed il pezzo o i pezzi con il maggiore attacco di giorno e di notte;
- il numero totale di pezzi sulla mappa;
- la cella o le celle con il maggior numero di pezzi dello stesso tipo.

### **Glossario:**

**File di input:** si tratta del file dal quale il programma, in fase di esecuzione, legge i dati che devono essere poi elaborati;

**Standard input:** lo standard input è un canale da cui giunge un flusso di dati (spesso testuali) che devono essere elaborati da un programma. In questo caso il flusso di dati è collegato alla tastiera dell'utente;

**Inizializzare:** significa assegnare un valore iniziale ad un determinato campo;

**Lista:** la lista o sequenza è una collezione di dati tra i quali esiste una relazione posizionale lineare, cioè di tipo precedente-successivo. Non c'è nessun requisito di ordinamento dei dati in base al loro valore, non è nemmeno necessario che i dati appartengano ad un insieme dotato di una relazione d'ordine.

**Costruttore:** il costruttore è il metodo (cioè la funzione) di una classe il cui compito è proprio quello di creare nuove istanze, oltre ad essere il punto del programma in cui un nuovo elemento (quindi una nuova entità) viene creato ed è reso disponibile per l'interazione con il resto del sistema.

**Nomenclatura degli intervalli:** la notazione utilizzata per gli intervalli, nell'elencazione dei requisiti, è da interpretarsi in questo modo:  $X \in [a, b]$  significa che  $X$  assume valori compresi tra "a" e "b", estremi inclusi.

**Requisiti aggiornati all'ultima elicitazione: 25/04/2020**

**Requisiti funzionali:**

<b>Funzione</b>	Creazione mappa
<b>Descrizione</b>	Dimensionamento della mappa in base all'input.
<b>Inputs</b>	M numero di righe, N numero di colonne.
<b>Sorgente</b>	Standard input.
<b>Outputs</b>	Mappa di dimensione MxN.
<b>Destinazione</b>	Campi della mappa.
<b>Azione</b>	Si richiede all'utente d'inserire da tastiera la dimensione della mappa nel formato (M, N).
<b>Requisiti</b>	I valori di M ed N devono essere noti.
<b>Pre-condizioni</b>	None.
<b>Post-condizioni</b>	Nel caso in cui M ed N siano valori validi la mappa viene creata.
<b>Effetti collaterali</b>	Se $M \leq 0$ oppure $N \leq 0$ le dimensioni non sono considerate valide. In caso di input non valido si richiede il reinserimento dei valori di M ed N.

<b>Funzione</b>	Posizionamento pezzi
<b>Descrizione</b>	Si posizionano i pezzi sulla mappa di dimensione MxN creata.
<b>Inputs</b>	Coordinate di un pezzo (X, Y), tipo del pezzo (le coordinate sono da interpretarsi come cartesiane).
<b>Sorgente</b>	File di input.
<b>Outputs</b>	I pezzi validi risultano posizionati sulla mappa.
<b>Destinazione</b>	Struttura dati di elaborazione.
<b>Azione</b>	Si apre il file di input. Si legge un pezzo dopo l'altro, secondo il formato specificato alla voce "Requisiti". Ad ogni lettura si posiziona il pezzo sulla cella della mappa in corrispondenza delle coordinate (X, Y) appena lette.
<b>Requisiti</b>	Affinché tutto vada a buon fine è necessario che il formato del file sia il seguente: X Y Tipo Ovvero, un pezzo è specificato da tre righe consecutive. Tra un pezzo e l'altro non ci devono essere ulteriori caratteri. I pezzi devono essere elencati in modo contiguo uno sotto l'altro. Il tipo è considerato valido se e solo se la prima lettera è maiuscola. Le coordinate (X, Y) di un pezzo possono essere qualsiasi, ma per essere considerate valide devono appartenere agli intervalli seguenti: ( $X \in [0, M-1]$ , $Y \in [0, N-1]$ ).
<b>Pre-condizioni</b>	Il file di input deve chiamarsi "pezzi.txt"
<b>Post-condizioni</b>	Chiusura del file.
<b>Effetti collaterali</b>	Se viene letto un pezzo le cui coordinate non rientrano nelle dimensioni massime della tabella, cioè $X \geq M$ oppure $Y \geq N$ tale pezzo non verrà inserito nella mappa. Se il file non è nel formato specificato alla voce "Requisiti" verrà visualizzato un errore.

<b>Funzione</b>	Capacità massima cella: 5 pezzi
<b>Descrizione</b>	Controllo del numero di pezzi in ogni cella.
<b>Inputs</b>	Numero di pezzi della cella.
<b>Sorgente</b>	Cella della mappa.
<b>Outputs</b>	Errore nel caso si ecceda la dimensione massima.
<b>Destinazione</b>	None.
<b>Azione</b>	Se nel posizionamento dei pezzi sulla mappa, una cella supera la dimensione massima consentita, si notifica il problema con un errore che richiede di modificare il file di input in corrispondenza delle coordinate interessate. Il programma infine termina.
<b>Requisiti</b>	File di input nominato "pezzi.txt", corretta formattazione del file.
<b>Pre-condizioni</b>	La mappa deve essere vuota.
<b>Post-condizioni</b>	Termine del programma.
<b>Effetti collaterali</b>	None.

<b>Funzione</b>	Conteggio pezzi fuori mappa
<b>Descrizione</b>	Si tiene conto dei pezzi che non si riesce a posizionare.
<b>Inputs</b>	Coordinate pezzo.
<b>Sorgente</b>	File di input.
<b>Outputs</b>	Stampa il conteggio per ogni tipo di pezzo.
<b>Destinazione</b>	Campi della mappa.
<b>Azione</b>	Se durante la lettura ad un pezzo di un certo tipo sono associate delle coordinate tali per cui $X \geq M$ oppure $Y \geq N$ allora tale pezzo non viene inserito nella mappa.
<b>Requisiti</b>	Il file di input deve essere nel formato corretto. Il tipo del pezzo deve essere uno dei tre accettati.
<b>Pre-condizioni</b>	I contatori devono essere inizializzati al valore 0.
<b>Post-condizioni</b>	Visualizzazione del numero di pezzi non validi distinguendoli per tipo.
<b>Effetti collaterali</b>	Potrebbe capitare che tutti i pezzi abbiano coordinate valide e che quindi non sono presenti pezzi fuori mappa.

<b>Funzione</b>	Tipologia pezzi
<b>Descrizione</b>	Si introducono quali sono le tipologie di pezzi della mappa e le loro caratteristiche. Ogni pezzo è un'entità indipendente dalle altre.
<b>Inputs</b>	Le tipologie di pezzo accettate sono 3. Ognuna di esse ha associato un valore di attacco ed uno di difesa espressi nel formato (Attacco/Difesa): Orco (4/4) Nano (2/5) Elfo (5/2)
<b>Sorgente</b>	File di input.
<b>Outputs</b>	Assegnazione dei valori di attacco e difesa.
<b>Destinazione</b>	Campi del pezzo.
<b>Azione</b>	Si associa ad ogni tipo di pezzo il valore corrispondente all'attacco ed alla difesa come definito alla voce "Inputs".
<b>Requisiti</b>	Solo i tre tipi di pezzi sopra specificati sono considerati validi.
<b>Pre-condizioni</b>	None.

<b>Post-condizioni</b>	None.
<b>Effetti collaterali</b>	None.

<b>Funzione</b>	Tipo non definito
<b>Descrizione</b>	Trattazione di pezzi con coordinate valide ma il cui tipo non è tra i tre ammessi.
<b>Inputs</b>	Tipo del pezzo.
<b>Sorgente</b>	File di input.
<b>Outputs</b>	Errore durante l'elaborazione.
<b>Destinazione</b>	None.
<b>Azione</b>	Se in fase di lettura dei dati si incorre in un pezzo di coordinate valide ma di tipo errato, cioè il tipo non corrisponde a nessuno dei tre descritti alla voce "Tipologia pezzi", il programma viene terminato notificando un errore nel file di input.
<b>Requisiti</b>	Solo i tipi Orco, Elfo e Nano sono considerati validi.
<b>Pre-condizioni</b>	Deve essere stato letto il tipo non valido.
<b>Post-condizioni</b>	Termine programma.
<b>Effetti collaterali</b>	None.

<b>Funzione</b>	Riferimento temporale
<b>Descrizione</b>	Associazione di una certa temporalità globale alla mappa
<b>Inputs</b>	Etichetta: "giorno" o "notte"
<b>Sorgente</b>	Scelta in fase di elaborazione.
<b>Outputs</b>	Dati elaborati in funzione del riferimento temporale.
<b>Destinazione</b>	Funzioni di elaborazione.
<b>Azione</b>	Si associa globalmente in fase di elaborazione dei dati un riferimento temporale alla mappa. Tale riferimento è assoluto quindi per ogni cella della mappa o è sempre "giorno" oppure è sempre "notte".
<b>Requisiti</b>	È necessario che solo uno dei due riferimenti temporali sia attivo, cioè o è "giorno" o è "notte". Non possono essere presenti contemporaneamente entrambi i riferimenti.
<b>Pre-condizioni</b>	Prima di iniziare ogni elaborazione assegnare al riferimento temporale il valore "giorno" e successivamente il valore "notte", oppure viceversa.
<b>Post-condizioni</b>	Elaborazione dei dati a seconda della temporalità.
<b>Effetti collaterali</b>	None.

<b>Funzione</b>	Tipologia cella
<b>Descrizione</b>	Associazione di una caratteristica tipologica alla cella.
<b>Inputs</b>	Etichette: pianura, bosco, montagna.
<b>Sorgente</b>	Generatore di numeri casuali.
<b>Outputs</b>	Tipo della cella selezionata.
<b>Destinazione</b>	Campi della cella.
<b>Azione</b>	Si crea un'associazione biunivoca tra i numeri 0,1,2 e le tipologie definite alla voce "input". L'associazione è la seguente: 0⇒bosco 1⇒pianura 2⇒montagna La generazione di una tipologia per ogni cella avviene casualmente.

<b>Requisiti</b>	None.
<b>Pre-condizioni</b>	None.
<b>Post-condizioni</b>	Sostituzione del valore numerico con il valore simbolico.
<b>Effetti collaterali</b>	None.

<b>Funzione</b>	Modificatori di combattimento
<b>Descrizione</b>	Modifica dei parametri di attacco e difesa in base alla tipologia della cella o al riferimento temporale della mappa.
<b>Inputs</b>	Tipologia della cella o riferimento temporale della mappa, valori di attacco e difesa del pezzo.
<b>Sorgente</b>	Dati della cella per il riferimento al tipo o dati della mappa per il riferimento temporale, dati del singolo pezzo per i valori di attacco e difesa.
<b>Outputs</b>	Valori di attacco e difesa modificati.
<b>Destinazione</b>	Campi del pezzo.
<b>Azione</b>	In base alla tipologia di cella su cui è posizionato il pezzo verranno modificati i suoi parametri di attacco e difesa, oppure la modifica dei valori di attacco e difesa dipende dal riferimento temporale della mappa. (I modificatori di attacco e difesa sono visualizzati nella tabella sottostante).
<b>Requisiti</b>	Per le combinazioni (tipo pezzo/tipologia cella) oppure (tipo pezzo/temporalità) per i quali non sono stati definiti modificatore di alcun genere non si applica alcuna modifica ai valori di attacco e difesa.
<b>Pre-condizioni</b>	Valori di attacco e difesa definiti nel requisito "Tipologia pezzi".
<b>Post-condizioni</b>	Valori di attacco e difesa modificati con i parametri percentuali sotto definiti.
<b>Effetti collaterali</b>	None.

I modificatori di combattimento sono visualizzati come degli incrementi o decrementi percentuali sul valore di attacco e/o sul valore di difesa del singolo pezzo.

<b>Tipo del pezzo</b>	<b>Tipo cella / Temporalità</b>	<b>Modificatore di attacco</b>	<b>Modificatore di difesa</b>
Elfo	Bosco	+0%	+100%
Nano	Montagna	+100%	+0%
Orco	Giorno	-50%	-50%
Orco	Notte	+50%	+50%

<b>Funzione</b>	Numero pezzi per tipologia
<b>Descrizione</b>	Conteggio del numero di pezzi presenti divisi per tipologia.
<b>Inputs</b>	Pezzi presenti sulla mappa.
<b>Sorgente</b>	File di input.
<b>Outputs</b>	Conteggio delle tre tipologie di pezzi.
<b>Destinazione</b>	Campi della mappa e visualizzazione sullo schermo.
<b>Azione</b>	Oltre a posizionare i pezzi sulla mappa si contano separatamente il numero di pezzi posizionati, quindi il numero di elfi, nani e orchi. Si visualizza poi sullo schermo il risultato.
<b>Requisiti</b>	È necessario leggere l'intero file.
<b>Pre-condizioni</b>	Azzeramento dei tre contatori, apertura del file.
<b>Post-condizioni</b>	Conteggio desiderato.

<b>Effetti collaterali</b>	None.
----------------------------	-------

<b>Funzione</b>	Maggiore difesa di “giorno” e di “notte”
<b>Descrizione</b>	Ricerca della cella in cui è posizionato il pezzo con la maggiore difesa quando il riferimento temporale è “giorno” oppure “notte”.
<b>Inputs</b>	Pezzi presenti sulla mappa.
<b>Sorgente</b>	Lista dei pezzi posizionati sulla mappa.
<b>Outputs</b>	Coordinate della cella con maggiore difesa quando il riferimento temporale è “giorno” oppure il riferimento temporale è “notte” (o le coordinate di più celle, vedere la voce “Effetti collaterali”).
<b>Destinazione</b>	Visualizzazione sullo schermo.
<b>Azione</b>	Quando il riferimento temporale è “giorno” oppure quando il riferimento è “notte”, si identifica la cella in cui è posizionato il pezzo con il valore di difesa più alto mediante ricerca lineare all’interno della lista dei pezzi posizionati sulla mappa.
<b>Requisiti</b>	None.
<b>Pre-condizioni</b>	I pezzi devono essere posizionati sulla mappa.
<b>Post-condizioni</b>	La mappa rimane inalterata.
<b>Effetti collaterali</b>	Potrebbe capitare che la cella individuata non sia unica. In caso di parità tra più celle verranno stampate tutte le coordinate delle celle che soddisfano la proprietà richiesta.

<b>Funzione</b>	Maggiore attacco di “giorno” e di “notte”
<b>Descrizione</b>	Ricerca della cella in cui è posizionato il pezzo con il maggiore attacco quando il riferimento temporale è “giorno” oppure “notte”.
<b>Inputs</b>	Pezzi presenti sulla mappa.
<b>Sorgente</b>	Lista dei pezzi posizionati sulla mappa.
<b>Outputs</b>	Coordinate della cella con maggiore attacco quando il riferimento temporale è “giorno” oppure il riferimento temporale è “notte” (o le coordinate di più celle, vedere la voce “Effetti collaterali”)
<b>Destinazione</b>	Visualizzazione sullo schermo.
<b>Azione</b>	Quando il riferimento temporale è “Giorno” oppure quando il riferimento è “Notte”, si identifica la cella in cui è posizionato il pezzo con il valore di attacco più alto mediante ricerca lineare all’interno della lista dei pezzi posizionati sulla mappa.
<b>Requisiti</b>	None.
<b>Pre-condizioni</b>	I pezzi devono essere posizionati sulla mappa.
<b>Post-condizioni</b>	La mappa rimane inalterata.
<b>Effetti collaterali</b>	Potrebbe capitare che la cella individuata non sia unica. In caso di parità tra più celle verranno visualizzate tutte le coordinate delle celle che soddisfano la proprietà richiesta.

<b>Funzione</b>	Maggior numero di pezzi dello stesso tipo
<b>Descrizione</b>	Identificazione della cella in cui si ha il maggior numero di pezzi tutti dello stesso tipo.
<b>Inputs</b>	Numero di elfi, nani e orchi di ogni cella.
<b>Sorgente</b>	Contatori di ogni cella.

<b>Outputs</b>	Coordinate della cella con maggior numero di pezzi dello stesso tipo (o coordinate di più celle, vedere la voce "Effetti collaterali").
<b>Destinazione</b>	Visualizzazione sullo schermo.
<b>Azione</b>	Si ricerca all'interno della mappa la cella in cui è presente il maggior numero di pezzi tutti della stessa tipologia.
<b>Requisiti</b>	None.
<b>Pre-condizioni</b>	I pezzi devono essere posizionati sulla mappa, ed i contatori delle tre tipologie di ogni pezzo devono essere inizializzati al valore 0.
<b>Post-condizioni</b>	La mappa rimane inalterata.
<b>Effetti collaterali</b>	Potrebbe capitare che più celle contengano lo stesso numero di pezzi dello stesso tipo e.g. una cella con 3 orchi, una cella con 3 elfi, una cella con 3 nani e una ulteriore cella con 3 orchi; in tale situazione si visualizzeranno le coordinate delle suddette celle poiché tutte rispettano la proprietà richiesta.

#### Requisiti non funzionali:

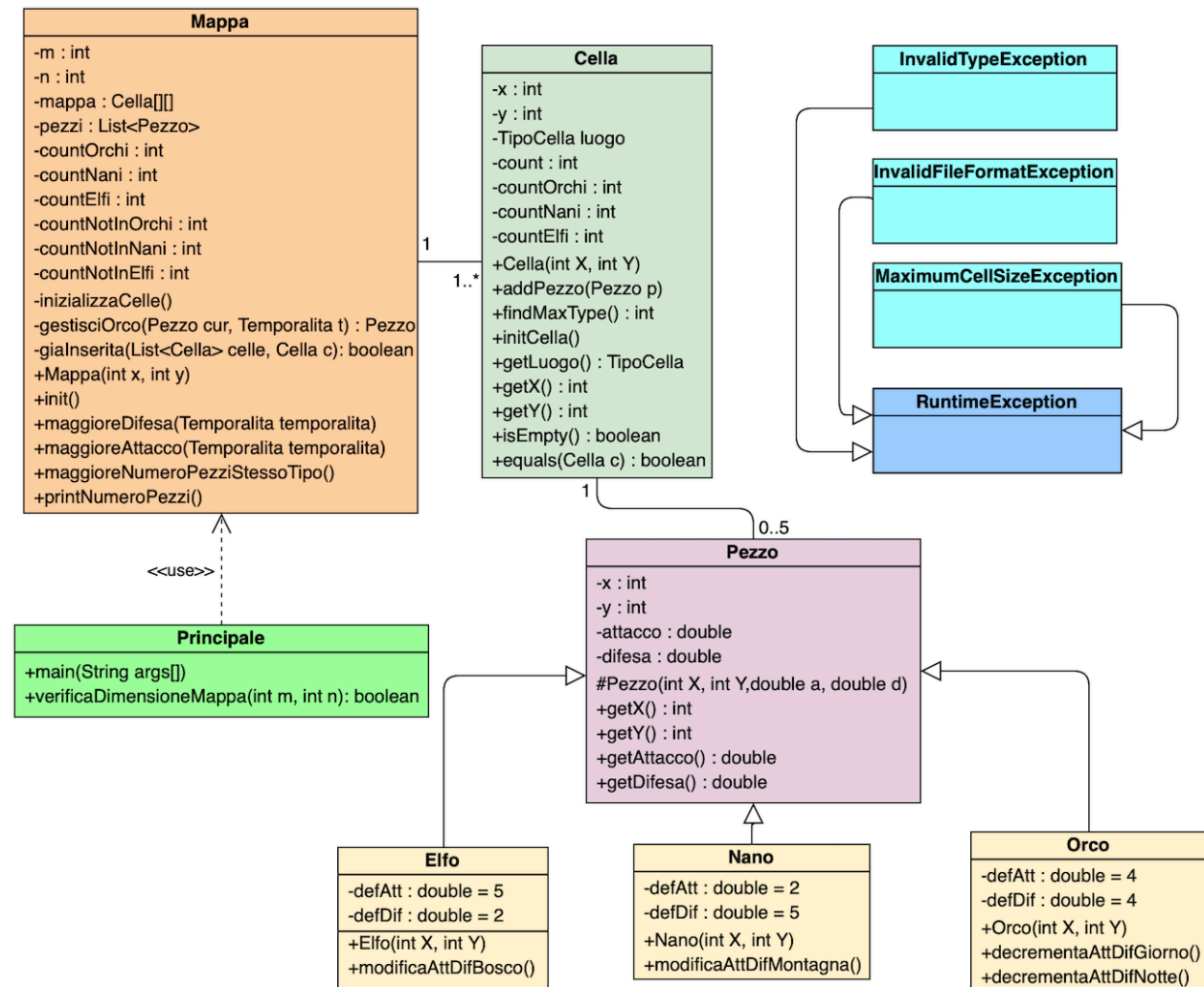
In tale sezione sono specificati i requisiti non richiesti espressamente dal customer ma che in fase di sviluppo hanno evidenziato la possibilità di creare problemi durante l'esecuzione del programma. Si è quindi deciso, dopo una consultazione con entrambi i committenti, di porre un limite superiore all'occupazione di memoria per evitare blocchi del programma nel caso della creazione di mappe di dimensione eccessiva.

<b>Funzione</b>	Dimensione massima mappa (500x500)
<b>Descrizione</b>	Definizione della dimensione massima della mappa
<b>Inputs</b>	M numero di righe, N numero di colonne.
<b>Sorgente</b>	Standard input.
<b>Outputs</b>	Mappa di dimensione MxN.
<b>Destinazione</b>	Campi della mappa.
<b>Azione</b>	Si verifica che i valori M ed N relativi alla dimensione della mappa appartengano agli intervalli seguenti: $M \in [1,500]$ , $N \in [1,500]$ .
<b>Requisiti</b>	I valori di M ed N devono essere noti.
<b>Pre-condizioni</b>	None.
<b>Post-condizioni</b>	Nel caso in cui M ed N siano valori validi la mappa viene creata.
<b>Effetti collaterali</b>	Se $M > 500$ oppure $N > 500$ allora le dimensioni non sono considerate valide. In caso di input non valido si richiede il reinserimento dei valori di M ed N.

#### Architettura software:

Per la parte architetturale del software si rimanda al "[Documento di architettura software](#)"

## System model (diagramma UML):



## System evolution:

Con dei possibili incrementi futuri si prevede la possibilità di poter introdurre nuovi pezzi all'interno del programma con relativi valori di attacco e difesa modificabili in determinate circostanze da determinare. Si prevede di poter introdurre nuove caratteristiche tipologiche per le celle e, funzionalità aggiuntive che permettono differenti elaborazioni dei dati. Si potrebbe magari rendere la mappa dinamica, cioè dopo un certo periodo di tempo, oppure a discrezione dell'utente, i pezzi potrebbero muoversi e cambiare cella e quindi potrebbero cambiare le loro coordinate. Allo stesso modo le celle potrebbero cambiare la propria caratteristica tipologica e così andrebbero ad influenzare i valori di attacco e difesa dei pezzi che sono presenti su di esse.

## Indice:

Introduzione .....	1
Glossario .....	1
Requisiti funzionali .....	2
Requisiti non funzionali .....	7
Architettura software .....	7
System model .....	8
System evolution .....	8