# PROGETTO POLINOMI

Classe Metodi

La classe contiene dei metodi ausiliari, ma fondamentali per la realizzazione dell’applicazione grafica.

* **static PolinomioLL inserimento( String linea ) :** Il metodo forma un PolinomioLL a partire da una stringa passata come parametro. Inizialmente se la stringa non soddisfa la regex tipica dei polinomi, viene invocato una IllegalArgumentException in quanto la stringa è malformata. Se essa supera il controllo, viene aperto un primo StringTokenizer, passando come delimitatori ‘+ -‘. Se un token è un operatore si modifica una booleana che indica se il polinomio è negativo. Altrimenti il token sarà un’espressione del tipo base^esp, quindi su di essa viene aperto un altro ST che divide la base e l’esponente per poter creare appunto il monomio da aggiungere al polinomio finale che verrà ritornato.
* **Static LinkedList<PolinomioLL> leggiFile( File F ) :** Il metodo serve per creare una LinkedList di Polinomi a partire da un file di testo. Può innescare la IOException in quanto si ha a che fare con file esterni, i quali, ad esempio, possono essere danneggiati. Viene aperto un BufferedReader sul file e viene scorso in tutte le sue righe. Infatti sulla linea viene aperto uno StringTokenizer con delimitatore ‘, ’, infatti i polinomi devono essere separati da una virgola o da uno spazio. Per ogni token si inserisce su una LinkedList(che verrà restituita) il polinomio creato a partire dal metodo precedente (inserimento(String linea)).
* **Static void salva( File f, LinkedList<PolinomioLL> lista ) :** Il metodo serve per salvare una serie di polinomi su un file di testo. Anch’esso può generare la IOException. Sul file viene aperto un PrintWriter, che scrive sul file ogni polinomio presente sulla LinkedList passata da parametro.

Classe PolinomioGUI

La classe è creata per implementare un’applicazione interattiva per lavorare con i polinomi. È composta da varie classi che corrispondono alle varie finestre alle cui si può accedere interagendo con la finestra principale.

* **class FinestraGUI :** Classe che compone la finestra principale dell’applicazione. Vengono impostate come si fa per tutte le finestre grafiche, dimensione, visibilità, ecc. Per quanta riguarda l’uscita viene mostrato un messaggio in cui si deve confermare la volontà prima di poter uscire. Viene creata la JMenuBar alla quale vengono aggiunti tutti i JMenuItem che richiamano le altre finestre grafiche, grazie ad un ActionListener, che sta alla base del funzionamento dell’interfaccia grafica.
* **class BoxArea :** Si accede cliccando nel menù principale il JMenuItem “Crea nuovo”. Finestra grafica all’interno della quale si inseriscono i polinomi attraverso una JTextField. Una volta inserito il polinomio si clicca su un JButton: Viene creato un polinomio a partire dal metodo inserimento() della classe Metodi. È passato come parametro il testo presente all’interno della JTextArea. Il polinomio appena generato viene inserito su una LinkedList.
* **class AddArea :** Si accede cliccando nel menù principale il JMenuItem “Addiziona”. Finestra grafica all’interno della quale è possibile selezionare i Polinomi aggiunti in precedenza e addizionarli, quindi mostrarne il risultato. I Polinomi vengono aggiunti al panello principale sottoforma di JCheckBox. I polinomi selezionati vengono aggiunti a una LinkedList. Tramite un JButton eseguo l’addizione tra i polinomi selezionati e mostro il risultato attraverso un messaggio informativo. Devono essere selezionati almeno due JCheckBox per effettuare l’addizione, in caso contrario viene mostrato un messaggio di errore.
* **class MulArea :** Si accede cliccando nel menù principale il JMenuItem “Moltiplica”. Finestra grafica all’interno della quale è possibile selezionare i Polinomi aggiunti in precedenza e moltiplicarli, quindi mostrarne il risultato. I polinomi vengono sempre aggiunti al pannello come in precedenza. Tramite un JButton eseguo la moltiplicazione tra i polinomi selezionati e mostro il risultato attraverso un messaggio informativo. Devono essere selezionati almeno due JCheckBox per effettuare la moltiplicazione, in caso contrario viene mostrato un messaggio di errore.
* **class DerivArea :** Si accede cliccando nel menù principale il JMenuItem “Derivata”. Finestra grafica all’interno della quale è possibile selezionare un polinomio ed effettuarne la derivata prima, quindi mostrarne il risultato. Tramite un JButton si esegue la derivate dell’unico polinomio selezionato e mostro il risultato tramite un messaggio informativo. Nel caso in cui i JCheckBox selezionati siano 0 o più di 1 viene mostrato un messaggio di errore.
* **class ValArea** **:** Si accede cliccando nel menù principale il JMenuItem “Calcola il valore”. Finestra grafica all’interno della quale è possibile selezionare un polinomio ed calcolarne il valore per una x specifica, quindi mostrarne il risultato. Viene aggiunto al pannello, oltre ai JCheckBox, un JTextField all’interno del quale inserire il valore della ‘x’ per cui calcolare il valore del polinomio selezionato. Tramite un JButton calcolo il valore del polinomio selezionato e mostro il risultato attraverso un messaggio informativo. Nel caso in cui i JCheckBox selezionati siano 0 o più di 1 viene mostrato un messaggio di errore.
* **class AscoltatoreEventiAzione :** Attraverso l’actionPerformered che è associato a ogni JMenuItem, viene aperta la finestra grafica corrispondente. I JMenuItem che non rientrano in questi casi sono:
* “Apri da file” : Viene aperto un JFileChooser. Una volta selezionato il file, la lista generale che contiene i polinomi (da cui si generano i JCheckBox) viene aggiornata con la lista prodotta dal metodo leggiFile della classe accessoria Metodi. In caso di problemi con l’apertura viene mostrato un messaggio di errore.
* “Salva con nome” : Viene aperto un JFileChooser. Si aggiorna quindi il fileDiSalvatggio e viene passato come parametro al metodo salva, della classe accessoria Metodi, che provvede a scrivere la lista generale dei polinomi fino ad ora inseriti, nel file assegnato.
* “Salva” : Se si è già effettuato un salvataggio in precedenza, esso aggiorna il file con le nuove modifiche apportate alla lista di polinomi.
* “About” : Si occupa di informare l’utente in maniera molto breve del funzionamento dell’interfaccia grafica.

Andrea Dell’Osso, mat n° 209865