

Programmazione 2

Prova Pratica - Durata: 2 ore

Check BST

Definire una funzione statica *checkBST()* che prende in input un nodo e verifica se esso è la radice di un BST i cui nodi sono stati inseriti correttamente. Si implementi una semplice classe BST, attraverso l'utilizzo dei template. Aggiungere alla classe BST un metodo di inserimento *inserisciMale()*, che inserisce un nuovo elemento nel BST violando la proprietà di ordine dei nodi di un BST.

Nel metodo main, istanziare un BST di interi ed un BST di char, successivamente inserire dei valori usando il metodo di inserimento tradizionale ed il metodo *inserisciMale()*. Successivamente testare la correttezza di un BST mediante il metodo *checkBST()* definendo sia esempi di BST corretti sia esempi di BST errati. Verranno valutati in maniera particolarmente positiva gli elaborati che definiscono *checkBST()* in maniera ricorsiva e che, una volta scoperto che il BST non è valido, indica all'utente quale sia il nodo posizionato erroneamente che ha generato l'anomalia.

Il codice deve essere fatto in modo tale da poter eseguire la compilazione ed il building su una macchina generica con il comando g++. Implementazioni "s sofisticate" comporteranno altrettante capacità di utilizzo del programma g++ da parte dello studente che le propone.