

## PROVA A - ALBERO BINARIO DI COMPUTER – 31/5/2018

Implementa un albero binario di ricerca NON GENERICO per memorizzare in modo ordinato un insieme di Computer ordinati per costo.

Implementa sia l'albero che la classe Computer in un unico file con nome tuo CognomeNome.java usando questo schema

*CognomeNome.java*

```
package cognomenome;

class Computer....{
....
}

class AlberoComputer {

....

}

public class CognomeNome{
    ... main(....){
        .....
    }
}
```

Un computer ha come campi solo il costo ed è Comparable rispetto ad esso.

L'albero deve avere i seguenti metodi:

- **insert** di inserimento di una automobile in ordine passato come argomento
- **print** che stampa il contenuto dell'albero in ordine
- **getMax** che resituisce il computer più caro.

Nel main fai un po' di prove in cui provi ad inserire programmaticamente qualche computer, stampare l'albero e stampa il massimo. Non fare alcun input con l'utente.

La consegna avviene tramite ilias nell'esercitazione sotto secondo compito:



## PROVA B PILA DI COMPUTER – 31/5/2018

Implementa una pila NON GENERICA per memorizzare con politica LIFO un insieme di Computer. Implementa la pila tramite liste dinamiche.

Implementa sia la pila che la classe Computer in un **unico** file con nome tuo CognomeNome.java usando questo schema

```
CognomeNome.java
package cognomenome;

class Computer ... {
....
}

class PilaComputer{
....
}
public class CognomeNome{
    ... main(...){
        .....
    }
}
```

Un computer ha come campi solo il costo ed è Comparable rispetto ad esso.

La lista deve avere i seguenti metodi:

- **push** di inserimento di un computer
- **pop** restituisce il computer (ultimo inserito)
- **find** di ricerca rispetto il costo (che restituisce true se esiste un computer con quel costo).
- **print** che stampa il contenuto della pila

Nel main fai un po' di prove in cui provi ad inserire programmaticamente qualche computer, stampare la pila e cercare computer. Non fare alcun input con l'utente.

La consegna avviene tramite ilias nell'esercitazione sotto secondo compitino:

The screenshot shows the Ilias submission page for the assignment 'Consegna prova B 31/5/18'. The page has a header with the assignment title and a sub-header with the instruction 'Consegnate qui l'unico file .java che avete sviluppato per la prova B'. Below this is a navigation bar with tabs: 'Assegnazioni', 'Info', 'Impostazioni', 'Richieste e Gradì', 'Progressi didattici', 'Esporta', and 'Permessi'. Under the 'Assegnazioni' tab, there are links for 'Visualizza' and 'Modifica'. A yellow banner indicates 'Consegna file (Obbligatorio)' with the note 'Remaining working time: Non e' specificata alcuna scadenza.'. Below this is a 'SCHEDULE' section with a table showing 'Remaining working time' as 'Non e' specificata alcuna scadenza.'. The 'LA TUA RICHIESTA' section shows 'File restituiti' as 'Non hai ancora inviato il file!' with a green 'Hand In' button.

## PROVA C MERGE SORT DI ARRAY DI Computer – 31/5/2018

Implementa un metodo (static) che faccia il sort di un array di Computer (Computer[]) mediante merge sort.

Implementa sia l'ordinamento che la classe computer in un unico file con nome tuo CognomeNome.java usando questo schema

**CognomeNome.java**

```
package cognomenome;

class Computer ... {
....
}

public class CognomeNome{

    static .... sort(...){ }

    .....

    ... main(...){
        ....
    }
}
```

Un computer ha come campi solo il costo ed è Comparable rispetto ad esso.

Il metodo di sort ordina l'array usando il metodo del merge sort.

Nel main fai un po' di prove in cui provi ad inserire programmaticamente qualche computer, stampare la lista. Non fare alcun input con l'utente.

Per stampare in contenuto dell'array usa il metodo [Java.util.Arrays.toString](#) che trasforma un array in stringa.

La consegna avviene tramite ilias nell'esercitazione sotto secondo compito:

The screenshot shows the Ilias submission page for the assignment 'Consegna prova B 31/5/18'. The page has a header with the assignment title and a sub-header 'Consegnate qui l'unico file .java che avete sviluppato per la prova B'. Below this is a navigation bar with tabs: 'Assegnazioni', 'Info', 'Impostazioni', 'Richieste e Gradli', 'Progressi didattici', 'Esporta', and 'Permessi'. The 'Assegnazioni' tab is active, showing a list of assignments. The first assignment is 'Consegna file (Obbligatorio)' with a status of 'Consegnato' and a remaining working time of 'Non e' specificata alcuna scadenza.'. Below this is a section titled 'SCHEDULE' with a table showing the remaining working time and the deadline. The table has two columns: 'Remaining working time' and 'Non e' specificata alcuna scadenza.'. Below the table is a section titled 'LA TUA RICHIESTA' with a table showing the status of the submission. The table has two columns: 'File restituiti' and 'Non hai ancora inviato il file!'. The 'File restituiti' column shows 'Non hai ancora inviato il file!' and the 'Non hai ancora inviato il file!' column shows a green button labeled 'Hand In'.