

Cognome e Nome _____

Matricola _____

Appello del 6 luglio 2022

Esercizio 1 [8 punti]

Si considerino le seguenti classi.

```
class A{
    int method(A a){
        return 10;
    }
    int method(B a){
        return 20;
    }
}
```

```
class B extends A{
    int method(A a){
        return 30;
    }
    int method(B a){
        return 40;
    }
}
```

Si dica cosa viene stampato a video dal seguente codice:

```
A a = new A();
B b = new B();
A c = new B();
System.out.println(a.method(a));
System.out.println(a.method(c));
System.out.println(b.method(b));
System.out.println(b.method(c));
System.out.println(c.method(a));
System.out.println(c.method(c));
```

Si giustifichi la risposta mostrando in particolare:

- le firme associate a tempo di compilazione e di esecuzione ad ogni chiamata di metodo
- l'evoluzione della memoria nella parte heap.

Esercizio 2 [6 punti]

- Mostrare l'**heap di minimo** che si ottiene a partire dall'heap vuoto inserendo nell'ordine indicato le chiavi **6, 32, 8, 45, 2, 80, 9, 7, 43** (mostrare l'heap che si ottiene dopo l'inserimento di ogni chiave);
- Rimuovere per 3 volte la chiave minima dall'heap, mostrando l'heap che si ottiene dopo ogni estrazione;
- Inserire nell'heap così ottenuto le chiavi **3, 72, 1**.

Regole per lo svolgimento della prova scritta:

- Per svolgere il compito si hanno a disposizione **120** minuti.
- Scrivere **subito** nome, cognome, matricola su **OGNI FOGLIO (compreso questo)**.
- Durante la prova scritta **non** è possibile abbandonare l'aula.
- Non è ammesso **per nessun motivo** comunicare in qualsiasi modo con altre persone
- È possibile consultare appunti, libri e dispense.
- Qualsiasi strumento elettronico di calcolo o comunicazione (telefoni cellulari, calcolatrici, palmari, computer, etc...) deve essere **completamente disattivato e depositato in vista sulla cattedra**
- Mettere in vista sul banco un valido documento di identità.

Esercizio 3 [9 punti]

- Si scriva una classe astratta **Temperatura** per rappresentare valori di temperatura
- La classe deve avere un metodo **public abstract double toKelvin ();** che sarà implementato dalle classi che estendono **Temperatura** con lo scopo di fornire una valutazione della temperatura in gradi Kelvin.
- La classe deve inoltre implementare un metodo **public boolean LEQ (Temperatura b)** che ritorna true se e solo se la temperatura su cui è invocato è minore o uguale a quella di b. *[si sfrutti il metodo toKelvin]*
- Si implementino infine le seguenti classi **concrete** che implementano **Temperatura**, in ognuna delle quali bisogna implementare un **costruttore** che prende un **double** e il metodo **toKelvin**:
 - La classe **TemperaturaCelsius** che possiede una variabile d'istanza di tipo double che memorizza la temperatura in gradi centigradi, ed il cui costruttore prende in input il valore della temperatura in gradi centigradi.
 - La classe **TemperaturaFahrenheit** che possiede una variabile d'istanza di tipo double che memorizza la temperatura in gradi Fahrenheit, ed il cui costruttore prende in input il valore della temperatura in gradi Fahrenheit.

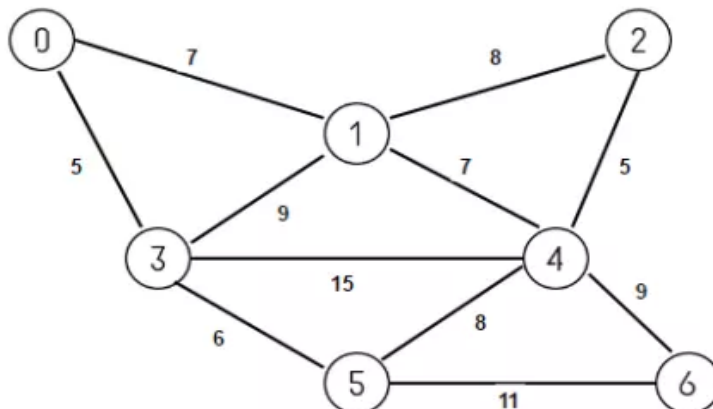
[Si ricorda che:

$$^{\circ}K = (^{\circ}F + 459,67) / 1,8$$

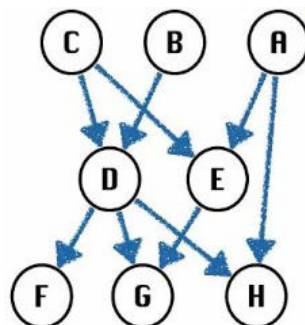
$$^{\circ}K = ^{\circ}C + 273,15$$

]

Esercizio 4 [11 punti]



- a. [6 punti] Mostrare una possibile esecuzione **passo-passo** dell'algoritmo di **Dijkstra** per i cammini minimi da singola sorgente, a partire dal nodo sorgente **6**. Ad ogni passo, bisogna mostrare il contenuto della codice con priorità utilizzata dall'algoritmo.



- b. [5 punti] Calcolare un ordinamento topologico usando il metodo della visita in profondità, evidenziando per ogni nodo i timestamp di inizio e fine visita. Per effettuare la visita in profondità, si parta sempre dal nodo non ancora visitato identificato dalla lettera che viene prima nell'ordinamento alfabetico.