Programmazione OO
Secondo Appello 17/02/2023

Non è consentito usare libri o appunti.

- **Esercizio 1** [**7 punti**] – Scrivere la classe `Pietanza` inserendola nel pacchetto `ristorante`. La classe implementa il comportamento di una pietanza disponibile in un ristorante. Una pietanza è costituita da un nome, da una lista di ingredienti e dal numero di calorie.
  La classe contiene i seguenti metodi:
  - `String dammiNome();`
  - `double dammiCosto();`

  Definire le classi `PietanzaFredda` e `PietanzaCalda`. La prima può essere un antipasto o un dolce. Nel primo caso il costo è di 8€ mentre nel secondo il costo è 6€. Una `PietanzaCalda` è caratterizzata da un tempo di cottura e da un peso. Il costo di una pietanza calda è 40€ al Kg. Inserire le due sottoclassi nel pacchetto `ristorante`.
  Scrivere la classe `Menu` che modella una collezione di `Pietanza` e inserirla nel pacchetto `ristorante`.

- **Esercizio 2** [**5 punti**] Scrivere la classe `PietanzaRichiesta` inserendola nel pacchetto `ristorante`. La classe implementa il comportamento di una pietanza ordinata da un cliente in un ristorante. Una `PietanzaRichiesta` è costituita da una pietanza, una quantità, e una variabile `servita`. Quest'ultima informazione indica se la pietanza è stata servita al cliente oppure no. Corredare la classe con opportuni metodi.

- **Esercizio 3** [**8 punti**] Scrivere la classe `Ordinazione` (nel pacchetto `ristorante`) che modella una collezione di pietanze. La classe è caratterizzata da un numero di tavolo e fornisce i seguenti metodi:
  - `void aggiungiPietanza(PietanzaRichiesta p)` che inserisce una pietanza ad un ordinazione in modo ordinato rispetto al nome. Nel caso in cui la pietanza `p` è già presente nell'ordinazione e non è ancora stata servita allora viene aggiornata solo la quantità della pietanza già presente. Nel caso in cui la `quantità` è negativa viene lanciata l'eccezione non controllata `QuantitaException`.
  - `ArrayList<PietanzaRichiesta> dammiProssimePietanze()` che restituisce la lista delle pietanze dell'ordinazione che non sono state servite;
  - `double dammiConto()` che restituisce il costo dell'ordinazione. Nel caso in cui l'ordinazione contiene pietanze non ancora servite lancia l'eccezione controllata `InvalidCostException`.
  - `void setPietanzaServito(int i)` che cambia lo stato dell'i-esima pietanza dell'ordinazione.

- **Esercizio 4** [**3 punti**] Scrivere una classe `Ristorante` caratterizzata da un menu ed una lista di ordinazioni. Corredare la classe con metodi per aggiungere, modificare e rimuovere ordinazioni.

- **Esercizio 5** [**7 punti**] Scrivere una classe di test che
  - legge dal file `Menu.txt` il menu del ristorante;
  - simula la creazione di 3 ordinazioni ognuna composta da 5 pietanze scelte in maniera casuale;
  - serva le pietanze ai clienti;
  - stampa il costo dell'ordinazione più economica.

Programmazione OO
Secondo Appello 17/02/2023

NOTE:

Ogni violazione delle regole enunciate ai punti sottoelencati comporta l'annullamento della prova (l'elaborato viene valutato 0).

1. Durante la prova d'esame è vietato usare:
    a. libri e appunti sia in forma cartacea che in forma digitale
    b. supporti di memoria esterni
    c. un font di dimensione maggiore di 10 punti.

2. Il nome del progetto consegnato deve cominciare con COGNOME seguito dal carattere underscore e quindi dal NOME (tutto in maiuscole). Ad esempio, il nome del progetto di Marco Rossi può essere ROSSI_MARCO, ROSSI_MARCO_P2, ROSSI_MARCO_ESERCIZIO, ROSSI_MARCO_549449384, etc.

3. Il file da consegnare deve essere creato da eclipse seguendo i passi:
    a. Seleziona "export…" nel menu file
    b. Seleziona "Archive File" in "General"
    c. Pressa "Next"
    d. Seleziona progetto da esportare
    e. Controllare il percorso del file (nell'area di testo con etichetta "To archive file:")
    f. Assicurarsi che i pulsanti radio nel pannello Options siano selezionati su "Save in zip format" e "Create directory structure for files"
    g. Pressa "Finish"

Assicurarsi che i progetti consegnati possono essere importati in eclipse come:

General → Existing Projects into Workspace