

Programma di INFORMATICA

Classe 4Ai

Unità 1 : IL LINGUAGGIO DI PROGRAMMAZIONE JAVA

Contenuto:

- Caratteristiche e applicazioni del linguaggio
- L'ambiente di sviluppo NetBeans
- Struttura di un programma Java e fondamenti del linguaggio
- La struttura di base di una classe
- Convenzioni di codifica del linguaggio Java
- Stringhe e codifica Unicode
- Gli array
- Gestione dell'input e dell'output
- La documentazione automatica dei programmi con Javadoc

Unità 2 : INTRODUZIONE ALLA PROGRAMMAZIONE E ALLA PROGETTAZIONE ORIENTATE AGLI OGGETTI

Contenuto:

- Le classi, gli oggetti e loro definizione (UML)
- Derivazione ed ereditarietà
- Polimorfismo e incapsulamento
- Analisi e sviluppo di elaborazioni utilizzando il paradigma object oriented

SEMESTRE

Unità 3. LA PROGRAMMAZIONE ORIENTATA AGLI OGGETTI IN JAVA

Contenuto:

- Definizione e generazione delle eccezioni
- Ereditarietà e polimorfismo
- Classi derivate; overriding e overloading dei metodi
- Classi astratte e interfacce
- Polimorfismo e binding dinamico
- Gerarchie di eccezioni e loro gestione

Unità 4. INTRODUZIONE ALLA GRAFICA JAVA (Laboratorio)

Contenuto:

- Studio dell'ambiente di sviluppo NetBeans con applicazioni Java con Swing
- Sviluppo di applicazioni Java con GUI Swing in ambiente NetBeans

Unità 5. GESTIONE DELLA CONCORRENZA NEL LINGUAGGIO JAVA

Contenuto:

- Thread in Java
- Condivisione di risorse tra thread
- Sincronizzazione dei thread

- Gestione dei socket

Unità 6 . OPERARE CON GLI ARCHIVI IN JAVA

Contenuto:

- Definizione di archivio
- Operazioni fondamentali sugli archivi
- Condivisione
- Tipi di archivio
- Tipi di organizzazione
- Tipi di accesso
- Trattamento dei file di testo in Java

Unità 7. STRUTTURE DATI DINAMICHE

Contenuto:

- ArrayList
- Concetto di lista lineare, pila e coda
- Concetto di albero binario

4Ai – Verifica di Informatica – 29/04/2016

Si deve gestire lo spoglio di una votazione, le schede devono essere memorizzate nel file testo schede.txt e deve essere possibile visualizzare il numero (anche parziale) delle schede bianche/nulle, dei voti di ogni candidato e il numero dei votanti

Il file schede.txt contiene i dati di una scheda in ogni sua linea:

“**” se la scheda è bianca/nulla oppure il nominativo del candidato se la scheda è valida

Per implementare l’elaborazione si deve sviluppare la classe Spoglio individuando gli attributi e i metodi:

- costruttore,
- inserimento(): inserisce i dati di una scheda nel file schede.txt
- calcolo(): calcola, utilizzando le schede presenti nel file schede.txt, il numero di votanti, i totali dei voti per ogni candidato e il numero di schede bianche/nulle.
- eventuali altri metodi che si ritengano necessari

Sviluppare un metodo main() che, utilizzando un oggetto della classe Spoglio, gestisca il seguente menù utilizzando i metodi inserimento() e calcolo().

1. Inserimento di una scheda nel file schede.txt
2. Calcolo: visualizza il numero delle schede bianche/nulle e i nomi e i voti dei candidati
3. Fine

L'elaborato deve contenere:

- UML della classe Spoglio documentato
- Analisi e flow con variabili documentate di tutti i metodi della classe Spoglio
- Analisi e flow del metodo main() con tabella variabili

NOTA: tutte le istruzioni sui file devono essere sviluppate nel flow con la codifica completa

Esercizio per mercoledì

Sia dato un file di testo CREDITI.TXT con il seguente tracciato logico:

Nominativo;Provincia;Importo

Le informazioni di ogni persona sono memorizzate su una linea
I dati sono ordinati per provincia come riportato di seguito.

Rossi;CL;1000
Bianchi;CL;256
Verdi;CT;100
Black;CT;990
White;CT;870
Neri;EN;450
Green;EN;200
Gray;EN;125
...

Memorizzare in un file TOTALI.TXT il totale degli importi per ogni provincia come indicato di seguito.

CL;1256
CT;1960
EN;325

4Ai - Verifica file, classi astratte, grafica – 1/4/2016 – Fila 1

Classe astratta

1. Sviluppare la classe astratta Animali, contenente due attributi di tipo String: nome e verso; un costruttore; un metodo astratto siMuove() che restituisce in una stringa dove si muove l'animale; un metodo astratto vive() che restituisce in una stringa dove vive l'animale; un metodo astratto chiSei() che restituisce in una stringa chi è l'animale; un metodo astratto mostra che restituisce una stringa contenente il nome+ chiSei()+verso +siMuove() + "vive"+ vive().
2. Sviluppare la classe Rana estensione di Animali

File

3. Dove si posiziona il puntatore del file in base alle modalità di apertura di un file?
4. Quando viene registrato l'EOF?

Grafica

5. Sviluppare un main() contenente il frame mioFrame di dimensione 450X500 con layout BorderLayout, creare il pannello mioPannello inserirlo sul frame nella zona South del layout del frame.

4Ai - Compito recupero insufficienti (oggetti) - 3/03/2016 1° trimestre Informatica

In un'azienda i dipendenti possono partecipare allo sviluppo di progetti (max 3)

Sviluppare la **classe Dipendente** formata dagli attributi nome, cognome ed un vettore vProgetti di oggetti Progetto contenente i progetti a cui partecipa il dipendente.

La **classe Progetto** è formata da: codice del progetto e descrizione

Prevedere i seguenti metodi della classe Dipendente:

- il metodo costruttore che riceve nome, cognome
- il metodo inProgetto() che inserisce un progetto nel vettore vProgetti[]
- il metodo toString

Prevedere i seguenti metodi della classe Progetto:

- il metodo costruttore che riceve il codice e la descrizione
- il metodo toString

Sviluppare la classe Test (contenente il metodo main()) che utilizza la classe Dipendente e Progetto.

L'elaborato deve contenere:

1. **Modello UML** delle classi con attributi, prototipi dei metodi e scopo
2. **Codifica della classe Dipendente** con analisi metodo inProgetto()
3. **Codifica della classe Progetto**
4. **Flow** della classe Test con metodo main() che utilizza i metodi delle classi

4Ai - Verifica recupero assenti e insufficienti ereditarietà – 26/02/2016

Il gestore di un negozio associa ad ogni **Prodotto** un *codice* a barre univoco, la *descrizione* del prodotto e il *prezzo* unitario. Ha inoltre la necessità di applicare il metodo *applicaSconto* del 5% sul prezzo del prodotto.

I prodotti sono suddivisi in **Alimentari** e **Non Alimentari**. Ai prodotti Alimentari viene associato *l'anno di scadenza*, mentre a quelli Non Alimentari il *materiale* principale di cui sono fatti. Realizzare le sottoclassi **Alimentari** e **NonAlimentari** estendendo opportunamente la classe Prodotto.

Nelle due sottoclassi specializzare il metodo *applicaSconto* in modo che nel caso dei prodotti

Alimentari venga applicato uno sconto del 20% se l'anno di scadenza è = 2016, mentre nel caso dei

prodotti Non Alimentari venga applicato lo sconto del 10% se il prodotto è composto da un materiale riciclabile come carta, vetro o plastica.

Realizzare una classe ListaSpesa con il metodo main() in cui si chieda all'utente di inserire i prodotti acquistati (codice, descrizione e prezzo) e calcoli il totale della spesa applicando gli sconti se il cliente ha la tessera fedeltà.

L'elaborato deve contenere:

1. Il modello UML delle classi Prodotto, Alimentari e NonAlimentari aggiungendo i metodi che si ritengono necessari
2. La codifica delle tre classi
3. La codifica della classe ListaSpesa contenente il metodo main(), indicando dove si verifica il polimorfismo e motivare.

Classe 4Ai - Verifica ereditarietà – 12/2/2016

1. la classe Robot, caratterizzata dalle proprietà stato (stringa) e velocità (double) e dai metodi: costruttore, setter/getter e toString
 - Sviluppare l'applicazione main, che crea istanze di Robot, testando i metodi. Il nome della classe con il main è UsaRobot.
2. la classe RobotVulcano, sottoclasse di Robot; in aggiunta ha la proprietà temperatura (double) e i metodi: costruttore, setter/getter e toString.
 - Sviluppare l'applicazione main, che crea istanze di Robot e RobotVulcano, testandone i metodi. Il nome della classe con il main è UsaRobotVulcano. Evidenziare un esempio di polimorfismo.
3. la classe RobotPalombaro, sottoclasse di Robot; in aggiunta ha la proprietà pressione (double) e i metodi: costruttore, setter/getter e toString.
 - Sviluppare l'applicazione main, che crea istanze di Robot e RobotPalombaro, testandone i metodi. Il nome della classe con il main è UsaRobotPalombaro. Evidenziare un esempio di polimorfismo.
4. la classe RobotLaser, caratterizzata dalle proprietà stato (stringa), velocità (double), temperatura (double) e potenzaLaser (double) e i metodi costruttore, setter/getter e toString. Definire la sua superclasse.
 - Sviluppare l'applicazione main, che crea un vettore con istanze delle quattro classi, testandone i metodi. Il nome della classe con il main è UsaTuttiRobot. Evidenziare un esempio di polimorfismo.

Dopo aver letto le richieste sviluppare:

- il modello UML delle classi sopra indicate esplicitando nel disegno la gerarchia dell'ereditarietà con le frecce.
- la codifica di tutte le classi.

4Ai - Compito recupero insufficienti (oggetti) - 28/01/2016
1°trimestre Informatica

In un'azienda i dipendenti possono partecipare allo sviluppo di progetti (max 3)

Sviluppare la **classe Dipendente** formata dagli attributi nome, cognome ed un vettore vProgetti di oggetti Progetto contenente i progetti a cui partecipa il dipendente.

La **classe Progetto** è formata da: codice del progetto e descrizione

Prevedere i seguenti metodi della classe Dipendente:

- il metodo costruttore che riceve nome, cognome
- il metodo inProgetto() che inserisce un progetto nel vettore vProgetti[]
- il metodo toString

Prevedere i seguenti metodi della classe Progetto:

- il metodo costruttore che riceve il codice e la descrizione
- il metodo toString

Sviluppare la classe Test (contenente il metodo main()) che utilizza la classe Dipendente e Progetto.

L'elaborato deve contenere:

5. **Modello UML** delle classi con attributi, prototipi dei metodi e scopo
6. **Codifica della classe Dipendente** con analisi metodo inProgetto()
7. **Codifica della classe Progetto**
8. **Flow** della classe Test che utilizza i metodi delle classi

Classe 4Ai - Verifica di Informatica- 27/11/2015 – Oggetti

Sviluppare una classe Distributore Biciclette che abbia i seguenti attributi:

- a. numero biciclette presenti utilizzabili
- b. numero biciclette presenti non utilizzabili perché guaste
- c. costo noleggio/ora
- d. importo totale incassato

Sviluppare un costruttore che inizializza, in modo parametrico, il distributore ad eccezione dell'importo totale incassato che sarà azzerato.

La classe dovrà prevedere la possibilità di:

1. **prelevare** una bicicletta tra quelle presenti e utilizzabili. Effettuare i controlli e gli aggiornamenti che si ritengono opportuni. Restituire l'esito dell'operazione.
2. **parcheggiare** una bicicletta (parametri: numero di ore di utilizzo, tipo di bicicletta guasta/aggiustata). Il metodo, dopo i controlli e gli aggiornamenti che si ritengono opportuni, calcola e restituisce l'importo da pagare.
3. **manutenzione** sostituzione delle biciclette guaste con biciclette utilizzabili
4. metodo **toString()**

Sviluppare una classe con il metodo main() TestDistributore che gestisce un distributore di biciclette utilizzando i metodi sopra indicati.

L'elaborato deve contenere:

1. Modello UML della classe con attributi e prototipi dei metodi con scopo
2. Analisi e flow dei metodi

3. Flow o codifica della classe TestDistributore contenente il metodo main()

Teoria:

Spiegare il significato di:

1. public e private
2. static

4Ai - Compito in classe di Informatica – 20/11/2015

Programmazione ad oggetti

Progettare la classe **Distributore**, per rappresentare un distributore di carburante. La classe ha i seguenti attributi:

- **pompadi** tipo double[] che contiene il quantitativo di benzina e di diesel disponibile nel distributore.
- **capacitaPompa** di tipo double[] che contiene la capacità di benzina e diesel del distributore.
- **prezzo** di tipo double[] che contiene il prezzo della benzina e del diesel, espresso in euro al litro.

I metodi della classe sono:

- il costruttore che riceve nei parametri il prezzo al litro della benzina e del diesel. Il distributore creato non contiene carburante.

La classe deve inoltre implementare i seguenti metodi:

- **rifornisci()** che rifornisce il distributore di benzina o di diesel
- **vendi()**, che vende una quantità di benzina o di diesel corrispondente all'ammontare di euro pagato, specificato dal parametro del metodo.
- **modificaPrezzo()** che aggiorna il prezzo della benzina o del diesel.

Nel metodo main, utilizzatore della classe, testare la classe e i metodi.

L'elaborato deve contenere:

1. UML documentato della classe Distributore
2. Analisi e flow di tutti i metodi della classe
3. Analisi e flow del main()

Compito in classe di INFORMATICA (classi-oggetti)

Risolvere i seguenti punti:

- 1 (1 punto) Definire che cosa si intende per classe e spiegare che relazione esiste tra classe e oggetti di quella classe.
 - 2 (4 punti) Dichiarare la classe Prodotto con gli attributi : codice, descrizione, prezzo e poi definire:
 - il costruttore di default
 - altri 3 costruttori che inizializzino lo stato dell'oggetto utilizzando i seguenti parametri:
 - costruttore a: int cod
 - costruttore b: String desc, float costo
 - costruttore c: int cod, String desc
 - 3 (5 punti) Dichiarare una classe di nome Casa (UML e codice) che ha:
 - come attributi privati il codice-casa, la superficie, il numero di vani, la via e il numero civico.
 - come metodi quelli necessari per soddisfare le seguenti richieste
 - a) la creazione di un oggetto attico di tipo Casa con
 - superficie = 150 , numero vani = 5, via = Po, numero civico = 2,
 - utilizzando un costruttore che permetta di fornire i valori degli attributi tramite parametri
 - b) un metodo per stampare tutti gli attributi dell'oggetto.
- Codificare la funzione main() che gestisce le richieste a) e b)

4Ai Compito in classe di INFORMATICA 3/11/2012 fila 2 (classi-oggetti)

Risolvere i seguenti punti:

- 1 (1 punto) Definire che cosa si intende per classe e spiegare che relazione esiste tra classe e oggetti di quella classe.
- 2 (4 punti) Dichiarare la classe Libro con gli attributi : codice, titolo, editore e poi definire:
 - il costruttore di default
 - altri 3 costruttori che inizializzino lo stato dell'oggetto utilizzando i seguenti parametri:
 - costruttore a: int cod
 - costruttore b: String tit, String ed
 - costruttore c: int cod, String tit
- 3 (5 punti) Dichiarare una classe di nome Mobile (UML e codice) che ha:
 - come attributi privati il codice mobile, il numero di piani, il colore, il tipo di legno e l'altezza come metodiquelli necessari per soddisfare le seguenti richieste
 - a) la creazione di un oggetto armadio di tipo Mobile con
 - altezza = 200 , numero di piani = 5, legno = faggio, colore = bianco,
 - utilizzando un costruttore che permetta di fornire i valori degli attributi tramite parametri
 - b) un metodo per stampare tutti gli attributi dell'oggetto.

Codificare la funzione main() che gestisce le richieste a) e b)