



LABORATORIO DI PROGRAMMAZIONE

Corso di Laurea Ing.
Gestionale 21/22
[Ing. Antonio Luca Alfeo](#)
luca.alfeo@ing.unipi.com

COME FARE IL TEST VIA MOODLE

1. Andare su <https://elearn.ing.uni.pi.it>
2. Tra gli esami cercare "Esami Informatica – CDL Ingegneria gestionale"
3. Fare Login in alto a destra con le credenziali di ateneo
4. Click test d'esame

The screenshot shows the Moodle course page for 'Esami Informatica – CDL Ingegneria gestionale'. The breadcrumb trail is 'Home / I miei corsi / Esami Informatica – CDL Ingegneria gestionale'. On the left, a 'Navigazione' sidebar lists the course structure, with 'Esami Informatica – CDL Ingegneria gestionale' expanded to show 'Partecipanti', 'Badge', 'Competenze', 'Valutazioni', and 'Introduzione'. On the right, a box contains a document icon with a checkmark and the text 'test 12 giugno 2022'. A blue arrow points from the 'Click test d'esame' instruction to this document icon.

N.B.: eventuali piccole variazioni della presente procedura saranno comunicate attraverso i canali istituzionali agli iscritti alla prova

ESEMPIO QUIZ E POSSIBILI RISPOSTE

Domanda **1**

Risposta salvata

Punteggio max.:
1,00

Con riferimento alla seguente istanza della relazione studenti:

Matricola	Cognome	Nome	Data di Nascita	Corso
243064	Rossi	Luca	null	Ing. Gestionale
243065	Bianchi	Luca	1999-04-23	Ing. Informatica
243066	Verdi	Marco	2000-05-15	Ing. Meccanica

Dire qual è il risultato della seguente query:

```
SELECT cognome FROM studenti WHERE year(datadinascita) = 1999;
```

- a. Le tuple Verdi e Rossi
- b. La tupla Verdi
- c. La tupla Bianchi
- d. Nessuna delle altre risposte

[Annulla la scelta](#)

Tempo rimasto 0:29:43

Domanda **2**

Risposta non ancora data

Punteggio max.: 1,00



Contrassegna domanda

Modifica domanda

quanto fa $3+2$?

- a. 2
- b. 1
- c. 5

← test quiz (nascosto)

Vai a...

test quiz (copia) (nascosto) ►

Termina il tentativo...

1. Questo e' un esempio di domanda del test
2. Solo una risposta per ogni domanda e' corretta
3. Notare che avete un tempo limitato per terminare il test
4. E' possibile rispondere alla domanda successive solo se avete risposto alla precedente

CONCLUSIONE DEL TEST

Al termine dell'ultima domanda del test sarà possibile cliccare su "invia tutto e termina"

Sarà richiesta la conferma dell'invio in quanto tale operazione non è reversibile

Vi saranno comunicate il numero minimo di risposte corrette per superare il test

Questo tentativo deve essere inviato entro venerdì, 27 maggio 2022, 08:59.

Invia tutto e termina

Conferma

×

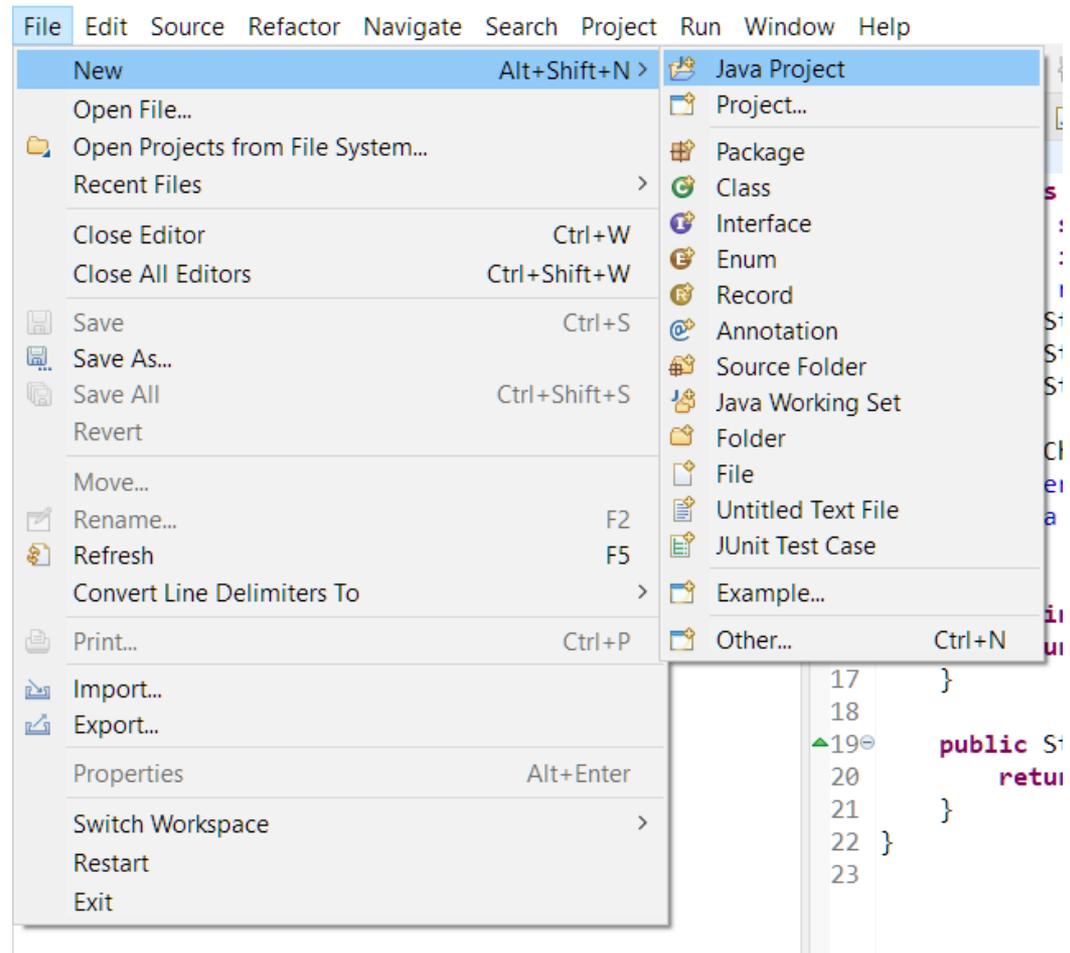
Stai per completare questo tentativo. Una volta chiuso il tentativo non potrai più modificare le risposte.

Invia tutto e termina

Annulla

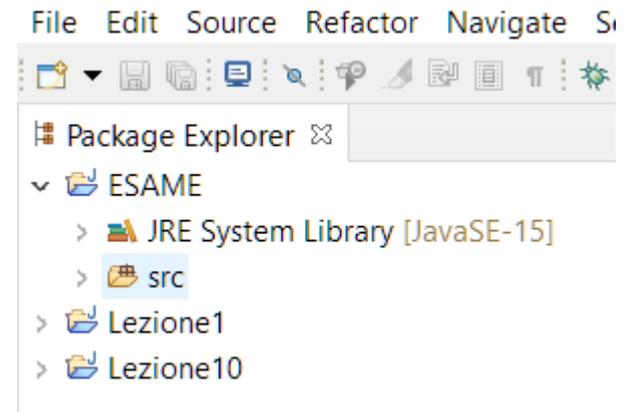
PROVA D'ESAME CON CLASSI FORNITE 1/3

1. Creare nuovo progetto
2. Nominare il progetto come "ESAME"
3. Selezionare "Don't create" quando viene richiesta la creare dei moduli

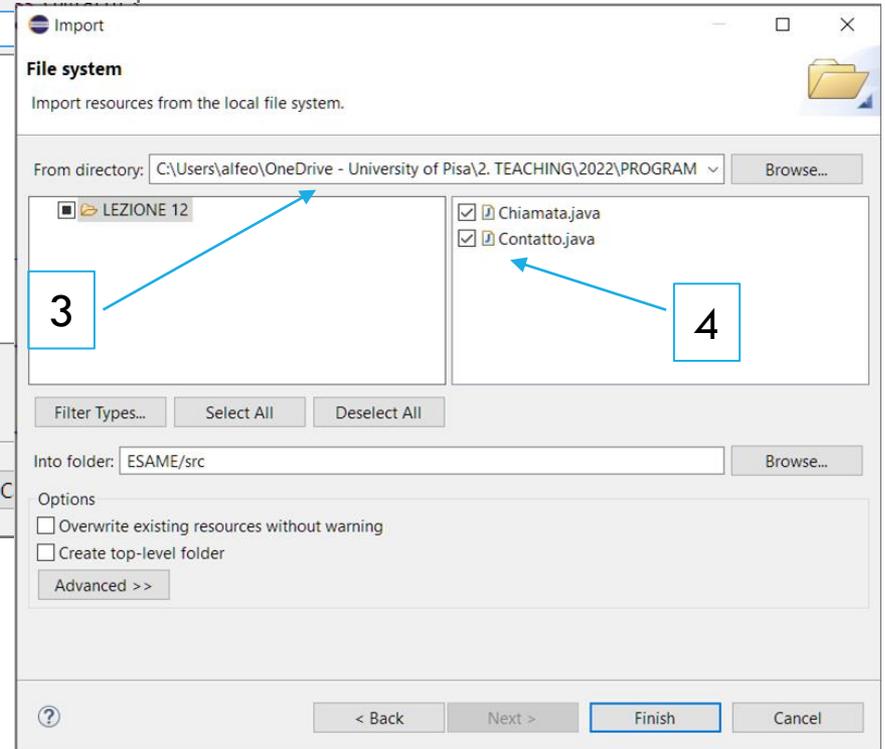
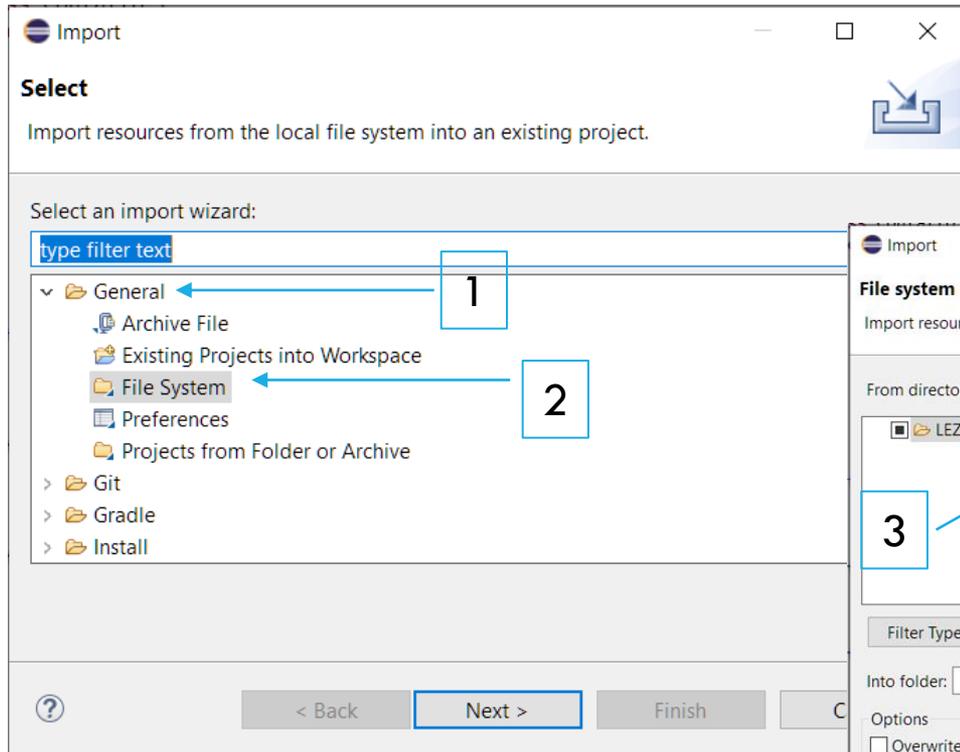


PROVA D'ESAME CON CLASSI FORNITE 2/3

1. Dal package explorer
2. ESAME → "src"
3. Tasto destro su "src" e `import`
4. Scaricare in una posizione nota i files `Chiamata.java` e `Contatto.java` da Teams, nel folder Lezione 12 del laboratorio



PROVA D'ESAME CON CLASSI FORNITE 3/3



CREARE NUOVA CLASSE

COPIARE IL MAIN



```
public static void main(String[] args) {  
    Contatto[] contacts = new Contatto[MAX_ELEM];  
    Chiamata[] calls = new Chiamata[MAX_ELEM];  
  
    contacts[0] = new Contatto(0, "Mario", "Rossi", "Piazza Cairoli, 3", "Pisa", "56124",  
        "050576904");  
    contacts[1] = new Contatto(1, "Pietro", "Rossi", "Via Roma, 31", "Pisa", "56127", "050642687");  
    contacts[2] = new Contatto(2, "Mario", "Ramarri", "Via Lenin, 4", "Roma", "00149", "0623476");  
    contacts[3] = new Contatto(3, "Giovanni", "Verdi", "Via Pollione, 4", "Chieti", "66100",  
        "08712278");  
    contacts[4] = new Contatto(3, "Giovanni", "Verdi", "Via Pollione, 4", "Chieti", "66100",  
        "08712278");  
    contacts[5] = new Contatto(1, "Pietro", "Rossi", "Via Roma, 31", "Pisa", "56127", "050642687");  
    contacts[6] = new Contatto(2, "Mario", "Ramarri", "Via Lenin, 4", "Roma", "00149", "0623476");  
    contacts[7] = new Contatto(2, "Mario", "Ramarri", "Via Lenin, 4", "Roma", "00149", "0623476");  
  
    calls[0] = new Chiamata(true, "01/01/2013", "00:00", "00:00:00");  
    calls[2] = new Chiamata(true, "03/02/2014", "09:45", "00:00:55");  
    calls[3] = new Chiamata(true, "02/04/2014", "10:21", "00:01:23");  
    calls[4] = new Chiamata(false, "07/07/2014", "15:32", "01:33:45");  
    calls[5] = new Chiamata(true, "09/10/2014", "17:04", "00:00:00");  
    calls[6] = new Chiamata(false, "09/11/2014", "07:45", "00:00:00");  
    calls[7] = new Chiamata(false, "25/12/2014", "12:03", "00:01:43");  
  
    System.out.println("\nA.");  
    System.out.print("Il numero totale delle chiamate perse è: ");  
    System.out.println(contaChiamatePerse(contacts, calls));  
  
    System.out.println("\nB.");  
    System.out.println("\nDatabase PRIMA dell'ordinamento:");  
    stampaDB(contacts, calls);  
    ordinaContatti(contacts, calls);  
    System.out.println("\nDatabase DOPO dell'ordinamento:");  
    stampaDB(contacts, calls);  
  
    System.out.println("\nC.");  
    System.out.print("Il costo totale dei tutte le chiamate effettuate è: € ");  
    System.out.println(costoChiamate(contacts, calls));  
  
    System.out.println("\nD.");  
    if (rimuoviContatto(contacts, calls, "0623476")) {  
        System.out.println("Le chiamate rimanenti sono:");  
        stampaDB(contacts, calls);  
    }  
    else{  
        System.out.println("Nessuna eliminazione.");  
    }  
}
```

TRACCIA D'ESAME DEL 15 GIUGNO 2015

Testo

Il database del sistema di gestione delle chiamate di uno Smart-Phone è costituito da due vettori paralleli. Il primo è denominato “contacts” e contiene oggetti di tipo “Contatto” che rappresentano i contatti presenti all'interno della rubrica dello Smart-Phone. Il secondo vettore è denominato “calls” e contiene oggetti di tipo “Chiamata” che rappresentano le informazioni relative ad ogni chiamata effettuata ad un determinato contatto. Ad ogni contatto può corrispondere più di una chiamata, in quel caso le informazioni del contatto saranno replicate.

Per ogni contatto presente nella posizione i -esima del vettore “contacts”, le informazioni relative ad una chiamata si troveranno nella corrispondente posizione del vettore “calls”. Nel caso in cui il contatto in posizione i -esima non abbia alcuna chiamata associata, nella posizione corrispondente nel vettore “calls” sarà presente un riferimento *null*. Entrambi i vettori hanno dimensione pari alla costante “MAX_ELEM” (inizializzata a 1024). Se il numero di contatti contenuti nell'archivio è inferiore a “MAX_ELEM”, i primi elementi del vettore conterranno gli oggetti di tipo “Contatto”, mentre gli altri conterranno riferimenti *null*. Tutti gli elementi *null* del vettore “contacts” si devono trovare alla fine del vettore e non possono trovarsi in mezzo agli elementi validi.

Le classe Contatto contiene le informazioni relative ad un contatto:

La classe Chamata contiene le informazioni relative alle singole chiamate ad un contatto.

Si consiglia di procedere implementando un metodo e successivamente la parte del main che utilizza tale metodo. Le varie operazioni devono essere eseguite sulla porzione significativa dell'archivio, cioè la porzione di “contacts” che non contiene riferimenti “null”.

A. Scrivere il metodo statico:

```
public static int contaChiamatePerse(Chiamata[] chiamate)
```

Il metodo deve contare il numero totale delle chiamate perse. Per chiamata persa si intende una chiamata ricevuta (attributo ricevuta = true, R/E = R nella tabella) con durata pari a “00:00:00”.

B. Scrivere il metodo statico:

```
public static void ordinaContatti(Contacto[] contatti, Chiamata[] chiamate)
```

Il metodo deve ordinare, nel vettore “contacts”, gli elementi in maniera crescente, usando come criterio la durata per elemento della chiamata corrispondente. Se la durata della chiamata è uguale vengono prima i contatti che hanno le chiamate ricevute. I contatti con chiamate nulle vanno in testa al vettore. Il metodo deve mantenere la corrispondenza iniziale tra contatti e chiamate. Se si ha la necessità di convertire una stringa in intero, si può utilizzare la funzione di libreria *Integer.parseInt(s)* che converte la stringa s in un intero restituito come risultato.

C. Scrivere il metodo statico:

```
public static double costoChiamate(Contacto[] contatti, Chiamata[] chiamate)
```

Supponendo un piano tariffario di € 0,003 al secondo senza scatto alla risposta, il metodo deve calcolare il costo totale di tutte le chiamate effettuate (attributo ricevuta = false, R/E = E nella tabella).

D. Scrivere il metodo statico:

```
public static boolean rimuoviContatto(Contacto[] contatti, Chiamata[] chiamate,  
                                     String tel)
```

Il metodo deve eliminare dal database, specificato dai parametri “contatti” e “chiamate”, tutti i contatti e le relative chiamate corrispondenti ad un determinato numero di telefono; restituire true o false a seconda del fatto che sia stato eliminato almeno un contatto e mantenere l'archivio in uno stato consistente.

E. Scrivere il metodo main che:

definisca ed inizializzi i vettori “contacts” e “calls” secondo i valori in tabella. La stampa dell'archivio consiste nello stampare le informazioni di ogni contatto e le chiamate associate (se ve ne sono). Si utilizzino correttamente i relativi metodi toString() implementati nelle due classi.

Id	Nome e Cognome	Indirizzo	Città	Cap	Telefono	R/E	Data (gg/mm/aaaa)	Ora (hh:mm)	Durata (hh:mm:ss)
0	Mario Rossi	Piazza Cairoli, 3	Pisa	56124	050576904	R	01/01/2013	00:00	00:00:00
1	Pietro Rossi	Via Roma, 31	Pisa	56127	050642687				
2	Mario Ramarri	Via Lenin, 4	Roma	00149	0623476	R	03/02/2014	09:45	00:00:55
3	Giovanni Verdi	Via Pollione, 5	Chieti	66100	08712278	R	02/04/2014	10:21	00:01:23
3	Giovanni Verdi	Via Pollione, 5	Chieti	66100	08712278	E	07/07/2014	15:32	01:33:45
1	Pietro Rossi	Via Roma, 32	Pisa	56127	050642687	R	09/10/2014	17:04	00:00:00
2	Mario Ramarri	Via Lenin, 4	Roma	00149	0623476	E	09/11/2014	07:45	00:00:00
2	Mario Ramarri	Via Lenin, 4	Roma	00149	0623476	E	25/12/2014	12:03	00:01:43

- Avvalendosi del metodo al punto A stampi a video il numero di chiamate perse.
- Ordini l'intero archivio utilizzando il metodo del punto B e stampi a video l'archivio prima e dopo l'ordinamento.
- Utilizzando il metodo C, stampi il costo di tutte le chiamate effettuate.
- Elimini il contatto con telefono 0623476, utilizzando il metodo del punto D. Al termine dell'operazione si stampi l'archivio aggiornato se l'operazione è avvenuta con successo, altrimenti si stampi un messaggio di errore.

SOLUZIONE

Su Teams, tra I files del laboratorio della Lezione 12 trovate il file:

[Appello_20150615.java](#)

Contenente una possibile soluzione alla prova d'esame.