

Concorso pubblico per esami per 1 posto Area dei Collaboratori - settore professionale tecnico, scientifico, tecnologico, informatico e dei servizi generali per il Dipartimento di Lingue e Letterature Straniere - Dipartimento di Eccellenza (cod. 2025cta005)

Bandito con D.D. n. 11603/2025 Prot n. 483211 del 31/10/2025, pubblicato sul Portale Unico del Reclutamento InPA e sul sito web di Ateneo in data 31/10/2025

Prova scritta - 10 dicembre 2025

Prova 1:

Per ogni risposta dei seguenti quesiti si riportino prima i comandi per eseguire le operazioni richieste e successivamente se ne spieghi e motivi l'utilizzo in modo conciso.

Quesito 1:

Facendo riferimento a un ambiente Linux standard su server remoto, accessibile via ssh tramite shell (bash o equivalente). Si assuma di essere correttamente connessi al server e di avere accesso a strumenti comuni come grep, head, tail, less, curl, netstat, nonché con utility per la manipolazione di file compressi (gzip, zcat, zgrep, tar).

Si assume di avere accesso ad un file di log molto grande (superiore a 1GB), compresso in formato .gz. Si riportino i comandi per verificarne nome, permessi e dimensione, e per estrarre solo le prime 50 righe del file che contengono la parola "ERROR". Se possibile si descrivano i comandi per ottenere il risultato senza decomprimere il file su disco.

Quesito 2:

Facendo riferimento a un ambiente Linux standard su server remoto, accessibile via ssh tramite shell (bash o equivalente). Si assuma di essere correttamente connessi al server e di avere accesso a strumenti comuni come grep, head, tail, less, curl, netstat, nonché con utility per la manipolazione di file compressi (gzip, zcat, zgrep, tar).

Si assuma che sul sistema sia installata una versione recente di Docker Engine, con i comandi docker disponibili da riga di comando e l'utente autorizzato a eseguire operazioni senza privilegi elevati (o tramite sudo).

Si supponga di aver avviato un container contenente un servizio web, utilizzando un comando del tipo:

```
docker run -d --name myapp -p 8080:80 myimage:latest
```

L'applicazione non risulta raggiungibile dal proprio browser. Occorre quindi verificare rapidamente se il problema dipenda da problemi di rete, dalla mappatura delle porte, dallo stato del container oppure da un errore interno dell'applicazione. Si descrivano i comandi utilizzabili e come interpretare il loro eventuale output.

Quesito 3:

Si consideri un ambiente Linux dotato di accesso a un sistema di gestione di database relazionali, ad esempio MySQL o PostgreSQL, utilizzabile tramite riga di comando mediante i rispettivi client standard (mysql per MySQL, psql per PostgreSQL).

Si assuma inoltre l'esistenza di una tabella denominata users definita come segue

```
CREATE TABLE users (
    id SERIAL PRIMARY KEY,
    name VARCHAR(100) NOT NULL,
    email VARCHAR(255) NOT NULL UNIQUE,
    verified BOOLEAN NOT NULL DEFAULT FALSE
);
```

Si richiede di redigere una query che selezioni esclusivamente gli utenti il cui stato di verifica risulta positivo, ordinandoli alfabeticamente in base al valore del campo 'name' e limitando l'output a un massimo di 20 righe.

Si forniscono inoltre, in maniera sintetica, le principali considerazioni da adottare qualora si intenda aggiungere alla tabella un ulteriore campo destinato alla memorizzazione della password degli utenti.

Quesito 4:

Si assuma un ambiente di sviluppo con Python 3 o PHP 8. Sono ammessi i seguenti costrutti del linguaggio:

- Tipi di base (interi, float, stringhe, booleani) e strutture dati semplici (liste/array, dizionari/associative array).
- Cicli (for, foreach, while) e condizionali (if/else, switch/case).
- Funzioni e metodi, con passaggio di parametri e valori di ritorno.
- Concetti di programmazione ad oggetti: classi, istanze, metodi pubblici/privati, proprietà, ereditarietà semplice.
- Non sono ammesse librerie esterne o framework particolari.

Si consideri una lista (o array) di stringhe contenenti codice HTML, ciascuna delle quali contiene esattamente e solamente un elemento <p>. Si richiede di sviluppare un programma che:

1. estragga il solo contenuto testuale racchiuso nei tag <p> e </p>, rimuovendo pertanto i tag HTML <p> ma mantenendo eventuali altri tag innestati;
2. concatensi i testi così ottenuti in un'unica stringa, separando i paragrafi mediante uno spazio semplice;
3. produca in output il risultato racchiuso all'interno dei tag <article> e </article>.

La soluzione deve contenere un esempio dell'input e un esempio dell'output e può essere realizzata sia secondo un approccio procedurale sia mediante programmazione orientata agli oggetti (ad esempio tramite una classe dotata di metodi appropriati).

Prova 2:

Per ogni risposta dei seguenti quesiti si riportino prima i comandi per eseguire le operazioni richieste e successivamente se ne spieghi e motivi l'utilizzo in modo conciso.

Quesito 1:

Facendo riferimento a un ambiente Linux standard su server remoto, accessibile via ssh tramite shell (bash o equivalente). Si assuma di essere correttamente connessi al server e di avere accesso a strumenti comuni come grep, head, tail, less, curl, netstat, nonché con utility per la manipolazione di file compressi (gzip, zcat, zgrep, tar).

Si consideri una directory contenente numerosi file di log compressi, tutti con estensione .log.gz.

Si richiede di descrivere come individuare la posizione in cui si trovano tali file ed estrarre, da ognuno di essi, le prime 20 righe che contengono la parola "ERROR".

Quesito 2:

Facendo riferimento a un ambiente Linux standard, accessibile tramite shell (bash o equivalente). Si assuma di avere accesso a strumenti comuni come grep, head, tail, less, nonché con utility per la manipolazione di file compressi (gzip, zcat, zgrep).

Si assume che sul sistema sia installata una versione recente di Docker Engine, con i comandi docker disponibili da riga di comando e l'utente autorizzato a eseguire operazioni senza privilegi elevati (o tramite sudo).

Si consideri un ambiente di sviluppo in cui si costruisce un'immagine Docker mediante un Dockerfile utilizzando un comando del tipo:

```
docker build -t myimage .
```

Durante la build, il processo termina con un errore “no such file or directory” durante l'esecuzione di un'istruzione `COPY` o `ADD`.

Si richiede di spiegare l'uso dei comandi COPY e ADD, e come individuare le cause più comuni che generano tale errore e di descrivere come risolverle. Si descriva poi il concetto di volume in questo contesto.

Quesito 3:

Si consideri un ambiente Linux dotato di accesso a un sistema di gestione di database relazionali, ad esempio MySQL o PostgreSQL, utilizzabile tramite riga di comando mediante i rispettivi client standard (mysql per MySQL, psql per PostgreSQL).

Si consideri una tabella orders composta dalle colonne *id*, *user_id*, *amount* e *created_at*, definita come segue:

```
CREATE TABLE orders (
    id SERIAL PRIMARY KEY,
    user_id INT NOT NULL,
    amount DECIMAL(10,2) NOT NULL,
    created_at TIMESTAMP NOT NULL,
    FOREIGN KEY (user_id) REFERENCES users(id)
);
```

Scrivere una query che, per un utente con id =42, calcoli il numero degli ordini con importo superiore a 100 e l'importo massimo tra quelli con importo superiore a 100.

Si forniscano inoltre, in maniera sintetica, le principali considerazioni da adottare qualora si intenda aggiungere una tabella destinata alla memorizzazione dell'indirizzo postale italiano degli utenti.

Quesito 4:

Si assuma un ambiente di sviluppo con Python 3 o PHP 8. Sono ammessi i seguenti costrutti del linguaggio:

- Tipi di base (interi, float, stringhe, booleani) e strutture dati semplici (liste/array, dizionari/associative array).
- Cicli (for, foreach, while) e condizionali (if/else, switch/case).
- Funzioni e metodi, con passaggio di parametri e valori di ritorno.
- Concetti di programmazione ad oggetti: classi, istanze, metodi pubblici/privati, proprietà, ereditarietà semplice.
- Non sono ammesse librerie esterne o framework particolari.

Si richiede di sviluppare un programma che, dato un array o una lista di URL:

1. Selezioni esclusivamente gli URL che iniziano con `https://` e terminano con `.jpg` o `.jpeg`, indipendentemente dal maiuscolo/minuscolo.
2. Producete come risultato una nuova lista/array contenente soltanto gli URL filtrati.

La soluzione deve contenere un esempio dell'input e un esempio dell'output e può essere realizzata sia secondo un approccio procedurale sia mediante programmazione orientata agli oggetti (ad esempio tramite una classe dotata di metodi appropriati).

Prova 3:

Per ogni risposta dei seguenti quesiti si riportino prima i comandi per eseguire le operazioni richieste e successivamente se ne spieghi e motivi l'utilizzo in modo conciso.

Quesito 1:

Facendo riferimento a un ambiente Linux standard su server remoto, accessibile via ssh tramite shell (bash o equivalente). Si assuma di essere correttamente connessi al server e di avere accesso a strumenti comuni come grep, head, tail, less, curl, netstat, nonché con utility per la manipolazione di file compressi (gzip, zcat, zgrep, tar).

Si assume di avere accesso ad un file di log molto grande (superiore a 1GB) , compresso in formato .gz. Si riportino i comandi per verificarne la locazione su disco e per spostarlo in una cartella /tmp.

Successivamente si verifichi se, nelle ultime 1000 righe, è presente almeno una occorrenza di una determinata stringa (ad esempio "INFO"). Se possibile si descrivano i comandi per ottenere il risultato senza decomprimere il file su disco.

Quesito 2:

Facendo riferimento a un ambiente Linux standard, accessibile tramite shell (bash o equivalente). Si assuma di avere accesso a strumenti comuni come grep, head, tail, less, nonché con utility per la manipolazione di file compressi (gzip, zcat, zgrep).

Si assume che sul sistema sia installata una versione recente di Docker Engine, con i comandi docker disponibili da riga di comando e l'utente autorizzato a eseguire operazioni senza privilegi elevati (o tramite sudo).

Si dispone del seguente Dockerfile:

```
FROM gcc:12
WORKDIR /app
COPY hello.c /app/hello.c
RUN gcc hello.c -o hello
CMD ["./hello"]
```

Si riportino i comandi necessari per eseguire la 'build' del container e per successivamente eseguire il comando.

Si descrivano i passaggi da effettuare nel caso si volesse poi modificare il codice sorgente del file 'hello.c' e testarne l'esecuzione. Si assuma che il comando necessiti di accedere a dei file all'interno di una cartella `/data`

Quesito 3:

Si consideri un ambiente Linux dotato di accesso a un sistema di gestione di database relazionali, ad esempio MySQL o PostgreSQL, utilizzabile tramite riga di comando mediante i rispettivi client standard (mysql per MySQL, psql per PostgreSQL).

Si consideri un database che gestisce pubblicazioni scientifiche con le seguenti due tabelle:

```
CREATE TABLE authors (
    id SERIAL PRIMARY KEY,
    name VARCHAR(100) NOT NULL,
    email VARCHAR(255) NOT NULL UNIQUE
);
```

```
CREATE TABLE publications (
    id SERIAL PRIMARY KEY,
    title VARCHAR(255) NOT NULL,
    year INT NOT NULL,
    journal VARCHAR(255) NOT NULL,
    author_id INT NOT NULL REFERENCES authors(id)
);
```

Si riporti la query SQL per calcolare il numero totale di autori che hanno almeno una pubblicazione su riviste il cui nome contiene la parola "Science" (case-insensitive).

Si forniscano inoltre, in maniera sintetica, le principali considerazioni da adottare qualora si intenda assicurare le migliori prestazioni per query simili a quella richiesta.

Quesito 4:

Si assuma un ambiente di sviluppo con Python 3 o PHP 8. Sono ammessi i seguenti costrutti del linguaggio:

- Tipi di base (interi, float, stringhe, booleani) e strutture dati semplici (liste/array, dizionari/associative array).
- Cicli (for, foreach, while) e condizionali (if/else, switch/case).
- Funzioni e metodi, con passaggio di parametri e valori di ritorno.
- Concetti di programmazione ad oggetti: classi, istanze, metodi pubblici/privati, proprietà, ereditarietà semplice.
- Non sono ammesse librerie esterne o framework particolari.

Si dispone di un array/lista di stringhe HTML, ciascuna contenente un tag `<h1>` o `<h2>` all'inizio della stessa seguito da del testo.

Si richiede di scrivere un programma che:

1. Conta quante stringhe contengono un tag `<h1>` e quante contengono `<h2>`.
2. Restituisca il conteggio sotto forma di dizionario/array associativo, ad esempio `{"h1": 3, "h2": 5}`.