

Esame Laboratorio di Programmazione I

22/06/2026

Svolgimento

Chiamate il file in cui scrivere il vostro codice `matricola.py` (la matricola è `SMnumeri` o `ECnumeri`).

All'inizio del file scrivete un commento con il vostro nome e numero di matricola.

Consegnate un codice che compila: se il codice non compila non sarà valutato.

Commentate il codice descrivendo in modo chiaro cosa state facendo.

Dataset Il file `ElectricityConsumption.csv` contiene una time series dei consumi mensili di energia elettrica, espressi in kWh.

Le colonne sono:

- `dt` (YYYY-MM-DD): la data della misurazione;
- `MinimumConsumption`: il consumo minimo mensile;
- `MaximumConsumption`: il consumo massimo mensile.

Il dataset può contenere dati mancanti, valori non numerici, date malformate oppure righe malformate con un numero di colonne inferiore a quello atteso.

1 (10 punti) Lettura dei dati

Scrivere la classe `CSVTimeSeriesFile`:

- (2 punti) La classe deve essere istanziata con il nome del file tramite la variabile `name`.
- (8 punti) Deve avere un metodo `get_data` che non prende input e restituisce una lista di triplette `[data, consumo_minimo, consumo_massimo]`, dove: il primo elemento è la data (stringa), il secondo è il consumo minimo mensile (float), ed il terzo è il consumo massimo mensile (float).

Devono essere restituiti tutti e soli i valori validi del file.

Esempio d'uso:

```
time_series_file = CSVTimeSeriesFile(name="ElectricityConsumption.csv")
time_series = time_series_file.get_data()
```

Esempio di output atteso:

```
[
    ["2018-01-01", 280.5, 390.2],
    ["2018-02-01", 275.0, 402.8],
    ["2018-03-01", 289.4, 410.1],
    ...
]
```

2 (12 punti) Calcolo delle variazioni annuali

Definire la funzione `compute_variations`, **NON** nella classe `CSVTimeSeriesFile` ma direttamente nel corpo principale del programma, che prende in input una serie temporale e un intervallo di anni:

```
compute_variations(time_series, first_year, last_year)
```

La funzione deve restituire un dizionario contenente, per ogni anno dell'intervallo selezionato, la variazione media annuale calcolata come:

variazione = media dei consumi massimi annuali – media dei consumi minimi annuali

La funzione deve:

- (3 punti) estrarre i valori minimi e massimi per ciascun anno;
- (3 punti) calcolare, per ogni anno, la media dei consumi minimi;
- (3 punti) calcolare, per ogni anno, la media dei consumi massimi;
- (2 punti) calcolare la variazione media annuale come `media_massimi - media_minimi`;
- (1 punto) restituire un dizionario contenente le variazioni arrotondate a 2 cifre decimali.

Esempio d'uso:

```
time_series_file = CSVTimeSeriesFile(name="ElectricityConsumption.csv")
time_series = time_series_file.get_data()
compute_variations(time_series, 2019, 2021)
```

Il dataset contiene valori dal 2018 al 2023, ma l'esempio considera solo gli anni dal 2019 al 2021.
Output atteso:

```
{
    "2019": 122.93,
    "2020": 112.37,
    "2021": 121.5
}
```

3 (8 punti) Validazione input ed eccezioni

Definire la classe:

```
class ExamException(Exception):  
    pass
```

e usarla per gestire i seguenti casi:

- (2 punti) Se il file non esiste oppure non è leggibile, si deve alzare un'eccezione che dice:

```
"Errore: impossibile leggere il file"
```

- (3 punti) Gli anni passati alla funzione devono essere interi, cioè di tipo `int`; altrimenti alzare un'eccezione che dice:

```
"Errore: l'anno inserito non è un intero"
```

- (3 punti) L'intervallo selezionato deve contenere almeno un anno con almeno un valore valido in output; altrimenti alzare un'eccezione che dice:

```
"Errore: l'intervallo selezionato non contiene valori validi"
```

Righe malformate, date malformate, valori mancanti o valori non numerici devono essere ignorati senza alzare eccezioni.