**Verifica di informatica – recupero**

Durata: 1 ora e mezza; aventi diritto: 2 ore

**Esercizio 1**

Si vuole realizzare in Python un programma per gestire liste della spesa. Il programma, organizzato a menù, deve consentire di:

1. aggiungere un nuovo articolo alla lista
2. eliminare uno specifico articolo dalla lista
3. visualizzare l’elenco degli articoli presenti nella lista
4. salvare l’elenco su file
5. caricare da file un elenco già fatto

Per il punto 4 occorre salvare i dati immediatamente ogni volta che l’utente apporta una modifica. Il file di testo deve avere estensione “.myl”.

Per il punto 5 il programma richiede il nome del file dal quale caricare i dati al momento dell’avvio (prima ancora di mostrare il menù principale). Per agevolare l’utente, il programma deve mostrare l’elenco di tutti i file “.myl” presenti nella cartella attuale e poi chiedere all’utente di indicare il nome del file dal quale si vuole caricare la lista. Se l’utente inserisce un nome presente in lista, i dati vengono caricati ed i successivi salvataggi andranno ad alimentare il file scelto; se il nome scelto non è presente in lista, allora verrà creato (ed alimentato) un nuovo file .myl

Oltre a sviluppare il software richiesto, organizzandolo in funzioni, lo studente implementi i dovuti controlli sui dati e si assicuri che le eccezioni vengano gestite correttamente.

**Esercizio 2**

Si vuole realizzare un programma Python che mostra frasi scherzose (jokes). Ogni frase può appartenere ad una sola delle seguenti categorie:

|  |  |
| --- | --- |
| **Categoria** | **type** |
| Generale | general |
| Toc-Toc! | knock-knock |
| Programmazione | programming |
| Papà | dad |

Le frasi vengono “generate” usando queste API:

* *https://official-joke-api.appspot.com/jokes/random* genera un joke random
* *https://official-joke-api.appspot.com/jokes/****{type}*** */random* genera un joke random di una certa categoria

Il programma deve mostrare un menù con due opzioni (più l’uscita)

1. mostra joke casuale
2. mostra joke di una categoria

Nel caso 2) occorre richiedere all’utente di quale categoria mostrare il joke, assicurandosi che venga scelta una categoria tra quelle ammesse.

Oltre a sviluppare il software richiesto, organizzandolo in funzioni, lo studente implementi i dovuti controlli sui dati e si assicuri che le eccezioni vengano gestite correttamente.

**Modalità di consegna**

Esclusivamente su piattaforma GITHUB.

**Tabella dei punteggi**

|  |  |
| --- | --- |
| **Voce** | **Punti** |
| Esercizio 1 | 50 |
| Esercizio 2 | 40 |
| Qualità codice  suddivisione in funzioni, commenti al codice, strutture dati utilizzate, implementazione algoritmi | 10 |
| **Totale** | **100** |