

Test finale della S2

Traccia:

Per agire come unHacker bisogna capire come pensare fuori dagli schemi. L'esercizio di oggi ha lo scopo di allenare l'osservazione critica.

Dato il codice si richiede allo studente di:

- Capire cosa fa il programma senza eseguirlo
- Individuare nel codice sorgente le casistiche non standard che il programma non gestisce.
- Individuare eventuali errori di sintassi/ logici.
- Proporre una soluzione per ognuno di essi.

La funzione "assistente_virtuale" é progettata per rispondere a comandi specifici relativi alla data, all'ora e all'identità dell'assistente. Essa utilizza il modulo "datetime" di Python per ottenere informazioni temporali e restituisce risposte predefinite in base al comando ricevuto dall'utente. Tuttavia nel codice originale erano presenti diversi errori che ne compromettevano il corretto funzionamento. Di seguito vengono elencati e spiegati tali errori, insieme alle proprie correzioni.

```
def assistente_virtuale(comando):  
    comando = "Qual'è la data di oggi?"  
    oggi = datetime.datetime.today()  
    risposta = "La data di oggi è " + oggi.strftime("%d/%m/%Y")  
    comando = "Che ore sono?"  
    ora_attuale = datetime.datetime.now().time()
```

Il primo errore é un **"Errore di Sintassi"** : il metodo utilizzato per ottenere la data corrente é scritto male, la sintassi corretta é "datetime.datetime.today"

```
def assistente_virtuale(comando):  
    comando = "Qual'è la data di oggi?"  
    oggi = datetime.datetime.today()  
    risposta = "La data di oggi è " + oggi.strftime("%d/%m/%Y")  
    comando = "Che ore sono?"  
    ora_attuale = datetime.datetime.now().time()
```

Il secondo oltre ad essere un errore di Sintassi, perché mancano i due puntini é anche un **Errore di Logica** perché i due puntini sono essenziali ad avviare il programma e senza di loro il ciclo sarebbe invalido, impedendo di proseguire.

```
def assistente_virtuale(comando):  
    comando = "Qual'è la data di oggi?"  
    oggi = datetime.datetime.today()  
    risposta = "La data di oggi è " + oggi.strftime("%d/%m/%Y")  
    comando = "Che ore sono?"  
    ora_attuale = datetime.datetime.now().time()
```

Gli errori di Fase di Esecuzione si verificano quando si usa esegue il programma e qualcosa va storto che non era evidente durante la fase di scrittura, corretti primi errori notavo che " L' assistente" non rispondeva alle domande e solo una volta levato il comando " break" riusciva a rispondermi, di per sé "break" non ha errori ma può causare problemi se se non viene utilizzato in correttamente.

```
File Azioni Modifica Visualizza Aiuto
GNU nano 8.1 esames215.py
import datetime
def assistente_virtuale(comando):
    if comando == "Qual'è la data di oggi?":
        oggi = datetime.datetime.today()
        risposta = "La data di oggi è " + oggi.strftime("%d/%m/%Y")
    elif comando == "Che ora sono?":
        ora_attuale = datetime.datetime.now().time()
        risposta = "L'ora attuale è " + ora_attuale.strftime("%H:%M")
    elif comando == "Come ti chiami?":
        risposta = "Mi chiamo Assistente Virtuale"
    else:
        risposta = "Non ho capito la domanda."
    return risposta
while True:
    comando_utente = input("Cosa vuoi sapere? ")
    if comando_utente.lower() == "esci":
        print("Arrivedesci!")
    else:
        print(assistente_virtuale(comando_utente))
```

Programma Corretto

```
File Azioni Modifica Visualizza Aiuto
GNU nano 8.1 esames215.py
import datetime
def assistente_virtuale(comando):
    if comando == "Qual'è la data di oggi?":
        oggi = datetime.datetime.today()
        risposta = "La data di oggi è " + oggi.strftime("%d/%m/%Y")
    elif comando == "Che ora sono?":
        ora_attuale = datetime.datetime.now().time()
        risposta = "L'ora attuale è " + ora_attuale.strftime("%H:%M")
    elif comando == "Come ti chiami?":
        risposta = "Mi chiamo Assistente Virtuale"
    else:
        risposta = "Non ho capito la domanda."
    return risposta
while True:
    comando_utente = input("Cosa vuoi sapere? ")
    if comando_utente.lower() == "esci":
        print("Arrivedesci!")
        break
    else:
        print(assistente_virtuale(comando_utente))
```

Programma Errato

Dopo aver corretto i diversi errori, il codice dovrebbe funzionare correttamente. Ora il programma può eseguire un ciclo continuo per ricevere input dall'utente e rispondere di conseguenza , terminando solo quando l'utente inserisce "esci"

