Trial exam – Example solution

November 22, 2022

Example solution programming

Task 1

1A The program starts by defining two lists, one for patients at the hospital and one for patients that have been discharged. It then prompts the user by input to enter the name of a patient. Depending on whether the patient is discharged or not, a proper message is given. If the patient is not in either list, an error message is given.

1B

- Input prompts the user to enter a text in the console. Input is always given as strings, so it is important to cast to a numeric type if working with numbers.
- Lists are used just like normal variables. The differene is that it can hold multiple values.

1C We can safely assume that a patient is unknown as we check both lists before entering the else-clause. Thus, we cover all logic posibilities apart from whenever the patient is not in either list.

Task 2

2A The value is 12.5

2B We use a for-loop as we know the number of iterations needed to perform the calculation. Using a while-loop would require an additional variable that has to be incremented for each iteration, which is implicit in a for-loop.

2C The variable `time_passed` is used as number of steps in the for-loop, and this parameter has to be a whole number. If we used a float the program would not be valid. We use float for `dosage` as this allows us to use a decimal number for the dosage of the drug.

Task 3

3A There are three main steps: - Open the file using `open("mood.csv")` - Creating an empty list - Using a for-loop to iterate through all lines using the `nextline()` function, storing each value as `int` in the list using `append`.

3B This algorithm assumes that we have stored the values in a list.

1. Iterate through the list and check each value
2. Create a variable `should_medicate = False`
3. For each step, check if the patient's score is above 7 or below 3. If either is true, change `should_medicate` to `True`
4. Print an appropriate message depending on the value of `should_medicate`

3C Check whether the value of the first element in the list is greater than the last element. Print an appropriate message depending on the if-statement.

Task 3 - code solution If you decided to write code instead of explaining, it could look like this:

```
# Task A
f = open("mood.csv")
moods = []
for line in f:
    moods.append(int(f.nextline()))

# Task B
should_medicate = False
for mood in moods:
    if mood > 7 or mood < 3:
        should_medicate = True
if should_medicate:
    print("The patient should be medicated")
else:
    print("The patient does not require medicine")

# Task C
if moods[0] > moods[-1]:
    print("The patient is more depressed")
else:
    print("The patient is more manic")
```