Team Application Exercises (tAPP-2) - Solutions

Problem 1

The code below should repeat the user's message the number of times that user wants. The code has 5 lines that are not in the correct order. Rearrange the code by entering the number that corresponds to each line of code.

Solution

3	`n = int(input("How many times should it be reapted? "))`
4	`for i in range(n):`
5	`print(message)`
2	`while message != "":`
1	`message = input("Enter a message (blank to quit): ")`

Problem 2

This code will determine whether or not a string entered by the user is a palindrome or not is not. The code has 11 lines that are not in the correct order. Rearrange the code by entering the number that corresponds to each line of code (whatever you think should be the first line of code will have number 1 in front of it).

Solution

6	`is_palindrome = False`
8	`if is_palindrome:`
10	`else:`
3	`i = 0`
5	`if line[i] != line[len(line) - i - 1]:`
1	`line = input("Enter a string: ")`
11	`print(line, "is not a palindrome")`
7	`i = i + 1`
2	`is_palindrome = True`
4	`while i < len(line) / 2 and is_palindrome:`
9	`print(line, "is a palindrome")`

Problem 3

When the following code runs, it asks for the name and then the age. If the age is under or equal to 10, the message is Hi followed by the name. The code has 13 lines that are not in the correct order. Rearrange the code by entering the number that corresponds to each line of code.

Solution

11	`def main():`

7	`if age <= 10:`
5	`return data_tuple`
8	`print("Hi", username)`
2	`username = input("Enter your user name: ")`
13	`message(username, age)`
1	`def get_data():`
6	`def message (username, age):`
3	`age = int(input("Enter your age: "))`
4	`data_tuple = (username, age)`
12	`username, age = get_data()`
10	`print("Hello", username)`
9	`else:`

Problem 4: Debugging code

The code should display a menu, add or subtract two numbers and display the corresponding messages. However, the code is not working. Identify the issues that affect this code.

Solution: There are 10 errors in total.

Line 11: attribute error – It should be randint.

Line 16: Name error – it should be answers (s at the end)

Line 20: Type error – it should be randint instead of random.

Line 23: Name error – it should be assignment operator instead of equality operator

Line 24: Syntax error – remove the colon

Line 28: Indentation error

Line 37: Syntax error –it should be equality operator instead of assignment operator

Line 38: UnboundLocalError – the local variable user_answer should not have the ‘s’ at the end.

Line 42: Synthax error – missing comma

Line 43: indentation error

```python
import random

def addition():
    num1 = random.randint(5,20)
    num2 = random.randint(5,20)
    print(num1, "+", num2, "= ")
    user_answer = int(input("Your answer: "))
    actual_answer = num1 + num2
    answers = (user_answer, actual_answer)
    return answers

def subtraction():
    num3 = random.randint(25,50)
    num4 = random.randint(1,25)
    print(num3, "-", num4, "= ")
    user_answer = int(input("Your answer: "))
    actual_answer = num3 - num4
    answers = (user_answer, actual_answer)
    return answers

def check_answer(user_answer,actual_answer):
    if user_answer == actual_answer:
        print("Correct")
    else:
        print("Incorrect, the answer is ", actual_answer)

def main():
    print("1) Addition")
    print("2) Subtraction")
    selection = int(input("Enter 1 or 2: "))
    if selection == 1:
        user_answer, actual_answer = addition()
        check_answer(user_answer, actual_answer)
    elif selection == 2:
        user_answer, actual_answer = subtraction()
        check_answer(user_answer, actual_answer)
    else:
        print("Incorrect selection")

main()
```

Problem 5:

This programme will ask the user to pick a low and high number, and then generate a random number between those two values and store it in a variable called “comp_num”. It will give the instruction “I am thinking of a number …” and then ask the user to guess the number they are thinking of.

`Incomplete function`: The third function should check to see if the `comp_num` is the same as the user’s guess. If it is, it should display the message “Correct, you win”, otherwise it should keep looking, telling the user if they are too low or too high and asking them to guess again (using a variable called `try_again`) until they guess correctly.

Task: Using a pen and paper, complete the function (with correct indentation).

Solution

```
import random

def pick_number():
    low = int(input("Enter the bottom of the range: "))
    high = int(input("Enter the top of the range: "))
    comp_num = random.randint(low, high)
    return comp_num

def first_guess():
    print("I am thinking of a number ... ")
    guess = int(input("What am I thinking of: "))
    return guess

def check_answer(comp_num, guess):
    try_again = True
    while try_again == True:
        if comp_num == guess:
            print("Correct, you win.")
            try_again = False
        elif comp_num > guess:
            guess = int(input("Too low, try again: "))
        else:
            guess = int(input("Too high, try again: "))

def main():
    comp_num = pick_number()
    guess = first_guess()
    check_answer(comp_num, guess)


main()
```