**Quiz 1 Mark Scheme**

How to use this mark scheme: The answers in red with the number of points in square brackets. Answers that are “close” but don’t work do not count because part A is designed to be short answers and students are allowed to use computers to test their code in part B. For some students, you might want to make sure their code exists on the computer they are using

**Part A: Short Questions [30]**

Answer these questions on this question paper. You are not allowed to use a computer for Part A: Hand in your answers to part A before doing part B

**Boolean (True/False) Questions [5]**

* + 1. 2 files can have the same file path on the same drive. False [1]
    2. Python is the only programming language. False [1]
    3. A script is used to write short commands and the shell is used to write longer programs. False [1]
    4. You must always import the math module before you use it. True [1]
    5. A function can have no parameters. True [1]

**Multiple Choice [5]**

* + 1. Which of the following rules (as many as you want) must you follow when naming variables?

1. The variable name cannot start with a number. True [1]
2. The variable name cannot have uppercase characters False [1]
3. The variable name should not be the same as that of a python keyword True [1]
4. The variable name can have spaces False [1]
5. The variable name can have any punctuation False [1]

**Short Answer Questions**

* + 1. What are the types of the values on the left? If it is not a valid type, write ERROR [7] ([1] each)

|  |  |
| --- | --- |
| 4567 | int |
| “345” | str |
| 324.423 | float |
| 5 >= 4 | bool |
| “Hello” | str |
| true | ERROR |
| None | None |

* + 1. Evaluate the following Boolean expressions. If it is not valid, write ERROR[7]

|  |  |
| --- | --- |
| 65 < 102 | True |
| 12 = 12 | ERROR |
| “haha” != “kkkk” | True |
| 1 or 0 | True |
| ( 3 == 7//4 ) and ( 3.4 is not 3 ) | False |
| True and False | False |
| ‘t’ in ‘Time’ | False |

* + 1. Identify the error in the following code blocks and debug (fix) it. [6]

#1

print(“Hello World”) Lower case p

#2

num = int(input(“Enter a number”)) Type cast num to an int/float

print(num + 7)

#3

print(“123” + “456”) Make 456 a string by adding “” or using str()

or

#3

print(123 + 456) Make 123 an int by removing “” or casting to int/float

#4

def myfunc(x):

return x + 2 Indentation

#5

s = “py”

y = s[2] Change to 0 or 1, index out of range

#6 Why does this not print anything in the script?

def hello():

print(“Hello”)

hello() call the function

**Part B: Coding Questions**

For the following questions you are allowed to use a script/shell to test your code but write the answers in the spaces given below.

**Shell Questions [4]**

* + 1. Calculate the following quantities in the shell and write the answer. Some of them are not written in Python Math! [4]

|  |  |
| --- | --- |
|  | 1050.192 |
|  | -0.404… |
| 17234 mod 332 | 302 |
| (2\*3%4)\*((1.4//0.4) + (15 - 3)) | 30.0 |

**Short Script Problems [4]**

* + 1. Define a function called **add2(***x,y***)** that adds 2 numbers *x* and *y* and returns the result*.*[2]

def add2(x,y):

’’’

Adds two numbers

x -> float

y -> float

returns float

’’’

return x + y [1] for return, [1] for x+y or equivalent

If they print instead of using return, give one point only

If they call the function, do not penalize them

* + 1. Define a function called **last\_char(***s***)** that prints the last character of a string *s* e.g. last\_char(“hello”) should print “o”, last\_char(“34”) should print “4”. [2]

def last\_char(s):

’’’

Prints the last character of the string s

s -> str

returns None

’’’

print(s[-1]) [1] for print, [1] for using proper indexing

if they return instead of print, give 1 point

if they call the function, do not penalize them

**Long Script Problem [12]**

You are going to write a script that asks for a user’s first and last name and then prints it in a specific format and tells them how many characters are in their name. Do not panic! It looks long but each step is very short. You do not need to define any functions.

You are allowed to ask for help on any of the questions but you will lose 1 point.

1. Create a new script and name it with your first name, last name and school (Remember the rules for naming files) and save it in your folder (e.g. C:/users/zimcode/My Documents/school/name).

**If you cannot do part a) ask for help, you will not be penalized.**

1. On the first line of your script write a comment with your name. [1]
2. Define/Declare a new variable called **first\_name**and make it input from the user with the question “First name? ”. [1]
3. Define/Declare a new variable called **last\_name**and make it input from the user with the question “Last name? ”. [1]
4. Replace the old **first\_name** with a new first name that is capitalized. (Hint: Take note of *replace* which means you must assign the capitalized first name to a variable!) [2]
5. Replace the old **last\_name** with a new last name that is all caps (upper case). (Hint: Take note of *replace* which means you must assign the uppercase last name to a variable!) [2]
6. Define a new variable called full\_name that is first\_name concatenated with last\_name with a space in between and print it. e.g. if first\_name is Tonde and last\_name is Moyo, full\_name should “Tonde Moyo” with a space in between.[2]
7. Store the length of full\_name minus one (for the space) in variable called name\_len. e.g. if the full name is “A B”, the length should be 2 not 3. [1]
8. Print the statement “Your name has *name\_len* characters” where name\_len is the answer from h) (Hint: What type should name\_len be for you to concatenate with a string?) [2]

Here are some examples of working code

Example 1

>>> First Name? Tonde

>>> Last Name? Moyo

Tonde MOYO

Your name has 9 characters

Example 2

>>> First Name? jerry

>>> Last Name? muleya

Jerry MULEYA

Your name has 11 characters

Example 3

>>> First Name? bABes

>>> Last Name? WodUMo

Babes WODUMO

Your name has 11 characters

# Name [1]

first\_name = input(“First name?: “) [1]

last\_name = input(“Last name?: “) [1]

first\_name = first\_name.capitalize() [1]replace, [1]s.capitalize()

last\_name = last\_name.upper() [1]replace, [1]capitalize()

full\_name = first\_name + “ “ + last\_name [1]concatenate, [1] add space

name\_len = len(full\_name) – 1 [1]use len()[1] subtract 1

print(full\_name) [1]

print(“Your name has”,name\_len,”characters”)[1]concatenate, [1] print

or

print(“Your name has ”+str(name\_len)+” characters”)