PRIFYSGOL ABERYSTWYTH - ABERYSTWYTH UNIVERSITY

DEGREE EXAMINATIONS 2016-2017 SEMESTER 2

INSTITUTE OF MATHEMATICS, PHYSICS AND COMPUTER SCIENCE

Computer Science, CS24420: Scientific Python

Time allowed: 1 hours

Calculators are permitted, provided they are either a Casio FX-83 or a Casio FX-85 device. No other calculators are permitted. Any calculator must be made available on request for inspection by invigilators, who are authorised to remove any calculators that are neither a Casio FX-83 nor a Casio FX-85 device.

This is a **mock exam**.
This is an **open book exam**.
Answer all questions.

Please write your answers in the answer template file (plain text), which can be downloaded from Blackboard: ⇒ Assignments ⇒ AnswerTemplate. The file name is *cs244mock-xxxx.txt*. Rename xxxx to your Aber Email ID. Include your answers, source code if applicable, but exclude any printing or plotting result. Submit your file via Blackboard.

1. This part consists of 2 questions without programming.

   a) Which of the following is true?   [1 marks]
      (1) In the hypothesis test, significance level = 1 - confidence level
      (2) In the hypothesis test, significance level = confidence level

   b) A person writes a piece of code to conduct a t test and calculate the pvalue as below. But the last line causes an error. How do you fix it?   [2]

```
import numpy as np
import scipy.stats as stats
population_mean = 0.5
sample = np.array([0,0,1,0,0,0,1,0,0,1,0,1])
result = stats.ttest_1samp(a = sample, popmean = population_mean)
pvalue = result['pvalue']
```

2. This part consists of 2 questions with programming.

   a) Write a Python program to calculate the center and spread of a ball in the *Lotto* draw history.

      1. Read the *Lotto* draw history, available at the Lotto website. Here is the link to a CSV data file:
         https://www.national-lottery.co.uk/results/lotto/draw-history/csv
         Alternatively you can download the data directly from Blackboard: ⇒ Assignments ⇒ DataSets. The file name is *LottoHistory.csv*.   [3]

2. Calculate the mean and standard deviation of *Bonus Ball* in the historical data. [4]

3. Boxplot *Bonus Ball* in the historical data. [3]

b) Write a Python program for dice roll simulation.

1. Calculate the mean of numbers on six faces of a dice, denoted by *popmean*. [2]

2. Throw a dice 100 times and record the outcome by a numpy array, denoted by *sample*. [5]

3. Conduct a one-sample t-test to test the null hypothesis: there is no difference between the *popmean* and sample mean. Set the significant level to 0.05. [5]