PX1224: Computational Skills for Problem Solving
Assignment 1

1. Vectors [2 marks]

The vectors x, y and z are given as:

x = −4.18 **i** +4.60 **j** +5.16 **k**
y = −3.47 **i** +3.40 **j** +1.07 **k**
z = −4.04 **i** +4.22 **j** +5.59 **k**

Of the four vectors:

a = + 5.46 **i** + 4.61 **j** + 3.06 **k** **b =** −5.12 **i** +5.82 **j** −2.24 **k**
c = +25.55 **i** −25.90 **j** −22.65 **k** **d =** −1.63 **i** +0.69 **j** +4.88 **k**

which one closest satisfies the following:

i. **Has the same length as x?[½]**

ii. **Is perpendicular to y?[½]**

iii. Lies in the same plane as x and z?**[½]**

iv. **Is parallel to y + z?[½]**

Please provide a python script that performs the required calculations and prints out the answers to the above questions. Note that the numbers in this question will not give exact results – provide answers that closest satisfy what is requested.

2. Statistics [2 marks]

We can model a coin toss as a binomial process with a probability p of getting a head. For example, **binomial(10, 0.5)** gives the number of heads you might get from 10 tosses of a fair coin (chance for heads is 50%, p = 0.5); **binomial(10, 0.5, N)** gives the answers you might get if you repeated the experiment N times.

i. **Generate 100,000 repetitions of 10 tosses of a fair coin, and make a fully labelled histogram of the results [with appropriate bins].[½]**

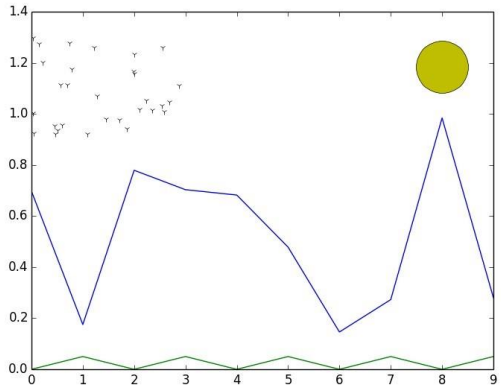
ii. **Calculate the mean and standard deviation of the number of heads.[½]**

iii. **In what fraction of trials do you get 7 or more heads in 10 tosses? (Hint: Week3 - 3.1.3 Truth testing on Arrays)** If you got 7 heads in 10 would you think the coin was unfair? Why?[½]

iv. **Now, generate 100,000 repetitions of 100 tosses of a fair coin. In what fraction of trials do you get 70 or more heads in 100 tosses?** If you got 70 heads in 100 would you think the coin was unfair? Why?**[½]**

3. Array manipulation and Plotting [2 marks]

The skills you have learned for making and manipulating arrays and formatting plots are very versatile. In less than 20 lines you should be able to make a code to produce a graph something like this:



Hints:

- **The grass is a simple array where every other number is zero. [½]**
- **The heights of the mountains are randomly generated.[½]**
- The sun and the birds are set to be above the height of the tallest mountain.
- The sun is a single point at x=8 with a very large circle marker.(Hint: search for marker size in Week 2)**[½]**
- The birds are the most complicated part. Their x-coordinates are randomly distributed between 0 and 3, and their y-coordinates are between the height of the tallest mountain and 1.3. (You can do this with `rand()`, or you might find it easier to use `uniform()` to generate uniformly distributed points in a given range). Use 'x' plot markers to represent the birds.**[½]**

4. Polynomials [4 marks]

A projectile is fired from an initial height of 90m, with a velocity of 120m/s at an angle of 40° above the horizontal. Assume g=9.8 m/s^2

a) Using `poly1d()` define polynomials x(t) and y(t) that describe the position of the particle as a function of time. (Hint: week3)

b) Using polynomial methods (Hint: roots, derivatives), find **[2]**

   i. The maximum height and the time at which the projectile reaches this height.

   ii. The time the projectile hits the ground and the horizontal distance it has travelled by this time.

c) Make an array of times between 0 and 15 seconds, with a spacing of 0.1 seconds between entries, then calculate arrays with the corresponding horizontal (distance) and vertical (height) positions. Use these to:**[2]**

   i. Find the maximum value in the height array, and the corresponding time from the time array. How does it compare with the exact answer calculated in (b)?

   ii. Find the range of times and horizontal positions for which the height is greater than 150m.(Hint: Week3 – 3.1.3 Truth testing)

5. Projectile motion and curve fitting [5 marks]

The motion of a ball is measured every 0.1 seconds for the first second of it's flight. The data is

Time(s)	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1.0
Height(m)	1.1	2.2	3.1	4.0	4.8	5.5	6.0	6.4	6.9	7.1

Write a code that fits a second order polynomial to the data (do not worry about calculating uncertainties on the coefficients). Use the code to:

i. Determine the maximum height the ball will reach **[1]** ii.

Determine the initial height of the ball (at t=0) **[1]** iii.

Determine the acceleration due to gravity **[1]** iv. Calculate

the time that the ball will hit the ground **[1]**

v. Generate a fully labelled plot showing the height of the ball against time, with both the data and the fit to the data. **[1]**

Ensure that your code is readable and well commented. Also, to be sure that you have everything you need in the code, make sure to restart the console and check that the code runs.

6. Straight line fitting [5 marks]

There is a file available with this assignment in Learning Central called **"skydive.txt"**. This contains noisy data giving the height above ground for a skydiver as a function of time. For this question, you are asked to provide a code that answers the following questions:

Write a code to do the following:

a) Read and make a plot of the data (Hint:Week 6 - 3.2.1) **[½]**

b) Estimate, by looking at the plot: **[½]**

   i. The time at which the skydiver reaches terminal velocity (with closed parachute)

   ii. The time at which the skydiver opens her parachute

c) Fit a first order polynomial to the appropriate data points to estimate the skydiver's terminal velocity, and the uncertainty in the value.**[1]**

d) Similarly, estimate the skydiver's speed, and the uncertainty in the value, with an open parachute.**[1]**

e) Assuming that the parachute takes 1 second to slow down the skydiver, what average acceleration does she experience during that time?**[½]**

f) Using the answers to part d, estimate the time at which the skydiver will land on the ground, with an estimate of the range of possible landing times.**[1]**

g) Make and save a plot showing the data and the curves that you have fitted to it.**[½] Please submit a Python code that performs the analysis described above:**

1. The code should make and **automatically save** a plot of the skydiver's position as a function of time, with the two best fit lines added
2. The code should produce an output text file that contains the answers to the questions above, given to an appropriate number of significant figures – clearly question a) does not need to be included in the output text file.