

Scenario: Creating a simple parser.

In engineering or data science, a routine procedure would require engineers to read files that are written in a particular format/style that is required to be parsed into another program for further processing. However, the input data may contain a lot of unwanted data! Your job is to "parse" only the relevant bit that is required for further processing. Take this example:

inputfile.abc

```
<?xml version="1.0" encoding="UTF-8"?>

<some other "useless" information goes here>

<name>Some Cool Data Follows</name>

  <pt alpha="1.23" beta="-6.78" gamma="70">

    </pt>

  <pt alpha="3.58" beta="-6.85" gamma="70">

    </pt>

  <pt alpha="4.69" beta="-6.82" gamma="NaN">

    </pt>

  <pt alpha="5.28" beta="-6.79" gamma="70">

    </pt>

  <pt alpha="6.33" beta="-6.81" gamma="70">

    </pt>

</end>
```

We can clearly glean from this file that the important bits of data are alpha, beta and gamma that are given in an ordered manner but in an unfamiliar format required for further processing. We require the data in the following form:

1.23	-6.78	70
3.58	-6.85	70
4.69	-6.82	NaN
5.28	-6.79	70
6.33	-6.81	70

Question 1: Using the example file above, copy and paste into notepad and save as inputfile.abc. Design and implement a C++ function to parse the required data as described above into an appropriate data structure of your choice. Your solution should also compensate for the existence of NaN's within the data set. (8 marks for C++ parser function, 4 marks dealing with NaN's) **[12 marks]**

Question 2: Write functions to generate two key performance indicators of your choice for each of the alpha, beta and gamma values.

(6 marks for C++ function and 2 marks for justification)

[8 marks]

Question 3: These data points (alpha, beta, gamma) actually represent points in 3D space where alpha is the x-axis, beta is the y-axis and gamma is the z-axis. Each entry represents the positional data (in kilometers) of a weather balloon relative to a weather station and taken every five seconds. The differential values (in each axis) give us an indication of the balloon distance away from the weather station and the speed of travel. Design and propose a C++ function to track speed and distance of the balloon. *Hint:* See https://www.engineeringtoolbox.com/distance-relationship-between-two-points-d_1854.html

(6 marks for distance measure, 2 marks for speed, 2 marks for justifications)

[10 marks]

Question 4: Provide a detailed flow diagram of your designed system.

[10 marks]

Question 5: Describe how you went about testing the system and provide your test results here.

[10 marks]

Type and submit a pdf file with your justifications and testing results only for **Questions 1 to 5** in the first submission area titled: Justifications and Test Results.

Submit your full C++ code in the second submission area. Additional marks shall be awarded for the following:

Proper use of functions (2 marks)

Use of exception handling (3 marks)

Proper use of comments (1 marks)

Appropriate data structure usage (4 marks)

[10 marks]

Total

[60 marks]