Assignment 1

- **Assignment release date: 09th January**
- **Assignment due date: 23rd January**
- **Submission mode:** via email with the title [DOE][Your Name]Assignment 1.
  You can include a typeset pdf (written in word or latex) or a scan of a handwritten document. As long as I can read and understand the contents it should be alright. If you have an accompanying code, please use a Google colab notebook and provide a link I can access

1. **Question 1** (10 points)
   As the manager of a newly developed laboratory, you have been tasked to set up a safety monitoring and alarm system that would buzz anytime a specific incident occurs more than a specified number of times. For one of the instruments, you have a device that records the number of operating conditions that exceed a set threshold in t seconds is counted. Your task is to devise a scheme such that the alarm will buzz if the number of incidents is above c. Based on your understanding of the instrument's safety manual, you have decided that if the frequency is less than 4, there should be a probability that the alarm will sound 1 in 10 times but will sound 9/10 times if the frequency is above 16. Assuming an appropriate distribution for this case, find out the numeric values for t and c.
   [Hint: This is a discrete distribution that we have not discussed in class]

2. **Question 2** (5 points)
   A study from a chemical reaction with an improved method B (from method A) has provided the following yield results. Apply the random sample and design methods to provide your conclusions about the experiment. Assume a similar data experimental design as the problem we worked out in class. Provide a python code for doing this in a Google colab.

Method A	Method B
54.6	74.9
45.8	78.3
56.3	68.1
50.7	66.5
64.5	73.5

3. **Question 3** (10 points)
   One of the important ideas in statistics is the *curse of dimensionality* which roughly translates to the scenario that as we move higher and higher in dimension, all the points tend to be very close in terms of a Euclidean distance between them. Can you think of a probabilistic way of showcasing this? Describe your method and provide a python code for doing this in a Google colab.