A big thanks to Adriana Chavez for assisting with the following description:

Many of us have seen how most of the computer applications can do a lot of calculations at a fast speed..But have you ever wondered what type of Math is needed in order to do such calculations?

With this presentation, I would like to show how some of the mathematical concepts used in areas of engineering and computer science.

**Discrete Mathematics:**

What is Discrete Mathematics?

First of all, “Discrete” Math is not a branch of Mathematics, like Calculus, algebra, etc. Instead, it describes a set of branches in Math that are “discrete” rather than “continuous”.

Most of the computer programs require some sense of “Logical equivalence”, and the use of sets and integers. Many Computer scientist had to learn how to reason with numbers; logical equivalence, mathematical proofs and induction. I believe these are the main components in writing a code. It is important that programs like C++, Phyton, Matlab, etc. run a code that is efficient enough to do a calculation for a finite amount of times.

Second, a very important part in Discrete Math is graph theory. Companies like Google and Yahoo first looked into graph theory when they started developing their algorithms. Graphs can be used to model relationship between objects and sets of objects. They are also used to model practically any algorithmic problem, and once you have defined the problem as a graph, you can solve it through graph operations like traversal or by checking for connectivity and circularity.

Discrete math is a broad term, but it was defined as a way to group the most important topics in math for needed for computer science. The more exposure a student has to these topics, the better they will be able to handle the challenges of software engineering.

**Linear Algebra:**

Linear algebra as a sub-discipline is often taught in one of two ways: from a computational aspect of things, which focuses on matrices, their properties, and operations on matrices; or, algebraically, where linear mappings are treated as algebraic structures, and one studies, for instance, the group theoretic relations that arise.

Linear Algebra is essential in Machine Learning, (Statistical Learning model). Here is a simple example. From what we know about linear algebra, we could use that information to say the following:

Machine Learning Support **Vector Machines** find a best separating hyperplane between two sets of vectors. The optimization problem minimizes an objective function that is most clearly expressed using linear algebra, the minimization algorithms are often solved in the dual space using linear algebra, and proofs regarding the algorithms involve linear algebra.

**Probability:**

This subject in math is the study of the odds that a particular event will occur. Probability plays a large role in how a computer program will work. The way in how an algorithm runs depends on multiple scenarios. (best-case, worst-case, average-case). Probability will help you determine the odds that the data coming into the algorithm is in a best or worst case ordering, and will guide you to your best solution.

A good software engineer or computer scientist questions everything when it comes to their design, and probability theory is an excellent tool in helping ask those questions. Probability is covered in high school statistics classes and there are many books and websites that explore it, often using dice rolls and decks of cards as examples.

-P 3.14-