**The Problem**

Congrats on mastering your first assignment with SF University. We hope you enjoyed building blocks and conquering the concept of poly blocks. Your new assignment is to create a block chain and explain how SF could benefit from its technology. We have provided some basic requirements that include payments for claims and documenting these transactions through block chain.

**Description of the Basic Requirements (Successfully pass JUnit tests)**

Block chain – A list of permanent records (blocks) that are linked together using a timestamp and other transactional data. The block chain eliminates paper processes and increases efficiencies building trust among all participants involved.

**BlockchainTest – step 1**

* Run JUnits 1st time – expected results (Runs: 3/3, Errors: 0, Failures: 3)
* Verify chain after adding 2 claims and 2 payments for those claims.
* Etc.

**Extra features - Step 2:**

Blockchain is not just for crypto currency. It has a number of other benefits. Please find features that could be useful from an insurance/financial perspective.

* Example of insurance implementation: <http://www.insurancejournal.com/news/international/2017/10/30/469647.htm>

**First Actions:**

* Import the problem statement into your IDE - GitHub
* Prepare for your presentation (No powerpoints – UML is a good way to present design)

**When you are done:**

* Update the feedback.txt file and include the following information:
  + Your team – name of each individual participating.
  + How many JUnits you were able to execute successfully.
  + Document and describe the additional “nice to have” features included, to help the judges properly grade your submission and explain how to properly execute new enhancements.
* Push your changes to one single branch for you and your teammate. Open a single pull request after development is completed.

**Questions:**

* Analysts will be outside of your room to help with any questions you may have.

**Rules**

* Contestants cannot seek help from individuals outside their team.
* Teams are expected to have the necessary tools and JARs preloaded on their machines **prior** to the competition.

**How you will be Graded**

Grading is broken up into 3 grading components:

Online Competition – 10

Finals Competition (coding) – 10

Presentation - 10

Finals Competition (coding): Points

* 100% core requirements (Step 1) met, including: 2
  + Number of JUnits that pass using correct functionality in the program
  + Code must compile and execute
* Code Cleanliness, Maintainability, Code documentation: 2
* Object-oriented principles: 2
* Creativity (Step 2): 4