**CSCI 310 Project**

**Group Assignment**

The purpose of this project is to increase your knowledge of the Bankers Algorithm,

Threads and synchronization by solving a problem using Java

**Members:**

Jared Rickert

Toua Vang

Sarbesh Banskota

**Theory:**

**Introduction**

The Banker's algorithm, sometimes referred to as the detection algorithm, is a [resource allocation](https://en.wikipedia.org/wiki/Resource_allocation) and [deadlock](https://en.wikipedia.org/wiki/Deadlock) avoidance [algorithm](https://en.wikipedia.org/wiki/Algorithm) developed by [Edsger Dijkstra](https://en.wikipedia.org/wiki/Edsger_Dijkstra) that tests for safety by simulating the allocation of predetermined maximum possible amounts of all [resources](https://en.wikipedia.org/wiki/Resource_(computer_science)), and then makes an "s-state" check to test for possible deadlock conditions for all other pending activities, before deciding whether allocation should be allowed to continue. [Wikipedia]

**Usage**

The Project uses multiple threads of execution that simulates the Banker Algorithm. Bank has a set number of resources to be provided to the customers and the Banker Algorithm is used to illustrate the thread management of that system. If a customer request is granted the customer then holds the resources until all of their needs have been met, then the customer returns all returns all resources to the bank. The possibility of Thread lock occurs if resources are granted in such a manner which prevents any customer from fulfilling the needs as the remaining other customers are holding resources that are needed. The algorithm is run by the operating system when a process request resources. It actually prevents deadlock if the accepting request puts the system in an unsafe state.

**User Manual:**

You can find the program on the csci2 where everything is already precompiled. You can find the class files in the target directory where main is defined on Program.class. You will also find a make file that simplifies building and running the program. The program will take a number for each resource type and a number for the number of customers. The program expects both numbers to be between 1 and 10. The program will then begin to log output about the current state of the program.

The address of the project is

* /export/home/cs310/cs310119/banker

To compile this program, simply enter:

* make build

The program is now compiled. To run the program enter this command:

* make run <number of resources> <number of customers> Ex. Make run 3 5