The main purpose is to realize Washall's Algorithm under the GPU Architecture using CUDA and to optimize the implementation, as well as compare the execution time of different approaches, which also includes a sequential program running on the CPU.

Washall's Algorithm also known as the Roy–Floyd algorithm is a fast method to compute the minimum distance between any two arbitrary vertices. The idea behind the algorithm is dynamic programming: if the minimal distance between any two vertices is known with the path only consist of a subset of the vertices, then we can add a new vertex and obtain the minimal distance with path containing the vertices by "relaxing" the distance of paths between all pairs with the newly added vertex. The correctness of it holds as long as there doesn't exist an negative circle. In a single iteration of k, edge ij only depends on edges ik and kj which are independent.

The Cuda threads are specified by kernels in the source, and are mapped to a maximum two three-dimensional space. The first three-dimensional space is the blocks. Each block is accessed through threadIdx and blockDim in the kernel. The second three-dimensional space is the grids. Each grid is accessed through blockIdx and gridDim in the kernel.

This implementation is the most straight forward way. Every time invoke the kernel that compute all ij for a specific k, as shown in the code. The value of k is enumerated in a loop in the host code. This program creates a thread for every element to be updated.The thread per block number is fixed at 1024.

CUDA provide an efficient way to program the GPU for general purpose calculations, which sometimes can be hundreds time faster and also offers powerful alternative in tackling massively-parallel problems.