RESEARCH ON

THREADMENTOR: MULTI-THREAD PROCESSING

RIJA FATIMA( 59631)

ARIBA SHAHID(59385)

HAMMAD HUSSAIN(59490)

(BSCS)

PAF-KIET

**ABSTRACT: (ARIBA SHAHID)**

This paper presents our effort in designing pedagogical tools for teaching message passing using channels. These tools include a class library that supports channels, a visualization system that helps students see the execution behavior of threads and message passing, and a topology editor that provides an environment for students to design network topologies. Moreover, since we have made sure the uniformity of the channel definition across the thread, parallel and distributed environments, porting a threaded program to a parallel/distributed environment is easy.

**OBJECTIVE (Hammad Hussain WORK)**

Thread Mentor is a multiplatform pedagogical tool designed to ease the difficulty in teaching and learning multithreaded programming. It consists of a C++ class library and a visualization system. The class library supports many thread management functions and synchronization primitives in an object-oriented way, and the visualization system is activated automatically by a user program and shows the inner working of every thread and every synchronization primitive on-the-fly. Events can also be saved for playback. In this way, students will be able to visualize the dynamic behavior of a threaded program and the interaction among threads and synchronization primitives.

**DESIGN (RIJA FATIMA WORK)**

Thread mentor consists of a class library and a visualization system. Both components support thread management (e.g., thread creation, termination, and join) and popular synchronization primitives (e.g., mutex locks, semaphores, monitors, barriers, and synchronous and asynchronous channels). The class library uses textbook syntax so that students do not have to memorize many different parameters, and hides as many system details as possible from its users.the class library translates many thread management and synchronization primitive calls to the corresponding system-supported thread library calls.

**Reference:**

<http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.418.6594&rep=rep1&type=pdf>