Assignment 2
Due 30 January 2007 (in class)


Problem 1: We would like to develop an algorithm for partitioning a streaming application, represented as a chain of computations with given computation workloads, for execution on a multi-core architecture. The objective is to minimize computation workload of the most intensive (largest) partition, which determines the application throughput.

a) Come up with an algorithm to partition the following sequence of computation workloads into three partitions: $4 \rightarrow 6 \rightarrow 10 \rightarrow 1 \rightarrow 3 \rightarrow 5 \rightarrow 7 \rightarrow 12$

b) Extend the idea to a method that works for an arbitrary sequence of workloads (integers), and targets an $n$-processor architecture.

Problem 2: The clique problem has been discussed and formulated using ILP in class. In this problem, you are asked to write a simple program to generate random connected graphs with 100, 1000, 10,000 and 100,000 nodes, and corresponding ILP problem instances in CPLEX format.

Random graphs should be created using the following method: 1) generate a graph with $n$ nodes and connect them to form a ring. 2) select a random-pair of nodes and insert an edge to connect them, if they are not already connected. 3) repeat step 2, until $10 \times n$ edges are inserted in the graph.

Solve the ILP instances with CPLEX, and report the solution time (reported by CPLEX) along with your code, problem instances, and solutions. For executing CPLEX on dept machines, you need to add the following line to your .software file:

cplex90

Then, you should be able to invoke cplex by just typing cplex. There are plenty of online manuals explaining the CPLEX input format, or alternatively, you can use its help/examples.