

# STiDC'06: Exercise 2

7th November 2006

## 1 Problem

The snapshot algorithm presented in the lecture has step complexity that is a function of the number of processes  $N$ . That is, in the worst case, a process needs  $f(N)$  steps to complete a single *update* or *scan* operation, where  $f$  is some function.

Imagine a situation in which  $N$  is very large, but usually only a few processes use a snapshot object. In such a scenario, it would be best to have a snapshot implementation which step complexity is not a function of  $N$ , but of the number of processes that use the shared object.

Your task is to write such an algorithm. More precisely, you should devise an algorithm for a (wait-free, atomic) snapshot object such that the step complexity of its *update* and *scan* operations is  $f(k)$ , where  $k$  is the number of processes that ever invoked either of the operations (in the current execution) and  $f$  is some function.

## 2 A Solution

A solution will be presented on the next exercise session and put on the course web page thereafter.