Australian National University
Department of Computer Science

Computer Science COMP3600/COMP6466 in 2007 – Lab 01

In this lab exercise, the problem is to find the kth smallest integer from a set of n integers.

Download the incomplete implementation `linear_select.c` of the linear-time selection algorithm. It can be found here:
`http://cs.anu.edu.au/student/comp3600/linear_select.c`

In this lab you need to finish the following tasks.

- Understand the source code of `linear_select.c`, and finish the incomplete part in the routine **selection**.
- Use the following data to verify the correctness of your algorithm. The input size is 19, and the input elements are

  39, 65, 25, 86, 35, 645,
  756, 24, 67, 98, 42, 68,
  96, 34, 5, 8, 74, 7, 245.

  The problem is to find the 8th smallest element from this set. The correct answer might be 42.
- Also check the program works for boundary cases like the smallest and largest elements.
- Investigate the claim that the time required by the algorithm is linear in the number of elements. To do this:
  - Change the program so that instead of reading the input data from the terminal, it just reads the number of elements n. Then it fills in the array $A[0..n-1]$ using calls to the library function `random()`. Then it calls `selection` to find the median element (the `(n+1)/2`-th element).
  - Make a measure of the running cost by adding a global variable which is initialised to 0 and incremented at the start of each function and inside each loop. At the end of the execution, print the value of this variable divided by n.
  - If the linear-time claim is valid, the value printed will be bounded regardless of the value of n. Run the program repeatedly with different values of n to see if this appears to be true.