# Foundations of Information Systems

Winter Semester 2023–24, Exercise 8

For discussion on Wednesday, December 20, 2023

1. Recall the "simple language" from class, which has non-negative integer variables, the statements `incr(X)` and `decr(X)`, as well as while loops of the form

   ```
   while(X):
       loop body
   ```

   In class, we looked at a macro `Y` $\leftarrow$ `X` which assigns the value of `X` to `Y`, while setting `X` to zero. Modify this example so that the end of the operation, `X` retains its original value.

2. Show that if a process occupies $m$ bytes of memory, virtual memory is divided into page frames of $s$ bytes each, and the page table entry for each page requires $t$ bytes of memory, then the optimal size of the page frame is

   $$s = \sqrt{2\,m\,t}\,.$$

   *Hint:* Assume that, on average, half of a page frame is wasted. Also assume that each page needs exactly one page table entry of $t$ bytes.[1]

3. Continuing Problem 2: On 32-bit Intel architectures, the standard size of a page frame is $4\,\text{KB} = 2^{12}\,\text{B}$. A single entry in the page table requires $32\,\text{bit} = 4\,\text{B}$. What is the implicit assumption on the typical size of a process?

4. Continuing Problems 2 and 3: On 64-bit architectures, the size of a page table entry goes up to $64\,\text{bit} = 8\,\text{B}$. How does the result change? Comment!

5. (Forouzan, P7-9.) Three processes ($A$, $B$, and $C$) are running concurrently. Process $A$ has acquired `File1`, but needs `File2`. Process $B$ has acquired `File3`, but needs `File1`. Process $C$ has acquired `File2`, but needs `File3`. Draw a diagram for these processes. Is this a deadlock situation?

[1] In practice, things are more complicated as page tables are organized as multilevel trees to be able to efficiently address non-contiguous blocks of virtual memory. So the formula given is just a back-of-the-envelope estimate.

6. (Forouzan, P7-10.) Three processes ($A$, $B$, and $C$) are running concurrently. Process $A$ has acquired `File1`. Process $B$ has acquired `File2`, but needs `File1`. Process $C$ has acquired `File3`, but needs `File2`. Draw a diagram for these processes. Is this a deadlock situation? If your answer is "no", show how the processes can eventually finish their tasks.

7. One solution to the *dining philosophers problem* demands that only four of the five philosopher sit at the table at any one time. When a philosopher finishes eating, she gets up and lets the standing philosopher take a seat. Is this solution free of deadlock? Is starvation possible?