UFAZ - Bachelor of Computer Science

System Programming

PW11 : signals

For each exercice, we expect the student to write a program, compile it and run it without errors of several examples. Test sets and comments are as important as the code itself.

Exercice 1

Write a program (an infinite loop) which increases a counter. When the user interrupts (signal `SIGINT`), the program records the date and the current value of the counter in a file, after what was already written. When the `SIGTERM` signal is received, the program write down `end` in the file and terminates.

We ask you to put all accesses to the file (open, write, close) in the functions associated to the signals. You shall use the `POSIX` primitives, making sure the handling of two signals do not create interferences.

Exercice 2

Redo exercice 1 by focusing on using *good practice* rules for signals, i.e. performing as few operations as possible in the functions associated to signals.

Exercice 3

We want to determine the size of a pipe.

To do that, we propose to send data, counting them, into a pipe which has no reader/listener (opened for reading, but never read by a process). After a while (a given number of bytes), the writer gets stuck waiting for the pipe to

empty itself. If a signal happens in this state, we can display the number of bytes in the pipe, i.e. its size.

Write a program which displays the size of a pipe using this method.