

# Solutions for Practice Problems for Week 1

## Operating and Systems Programming 2022/23

For all questions involving code, try to answer them *without* running the code first.

1. The first and second instructions use the I/O interfaces to store a value in the registers. The third instruction uses the ALU to compute the sum and stores the result in the register c. The fourth instruction stores the value of the register c in main memory. The fifth instruction again uses the ALU to compute the difference and stores in the register d. The final instruction uses the I/O interfaces to print the value of d.
2. Access to registers inside the CPU is much faster than main memory. You may save read and write-access to memory by using registers which is accessed frequently. The instructions given in the first question are an example: the registers a and b are used several times, saving memory access time.
3. Pointer variables store memory addresses.
4. Assume an array is declared as `int array[100]`. Then `array` may be used as a pointer to `array[0]`. Conversely, if you have allocated enough memory, you may use `*(p+i)`, where `p` is a pointer and `i` is an integer, for `p[i]`.
5. The program swaps the content pointed to by `px` and `py`.
6. 90.500000  
3
7. 2, 1
8. The string uses 13 bytes - 12 bytes for storing "Hello World!" (one byte per character) and one byte for the final `'\0'` character.
9. This function copies the string `t` to the string `s`, disregarding any possible problems with memory allocation.
10. This function does the same as the previous one. It uses pointer arithmetic instead of an explicit index.