

Quiz 1

The Art of Computer Programming

21 September 2026

Full name:

You have 10 minutes to answer this quiz, directly on this sheet of paper. No electronic devices or material of any kind is allowed. Do not forget to add your name above. Every question amounts to 2 points; the quiz is graded out of 10 points.

Questions

Q1. Circle the (only) correct answer. With the tools used in this course (GCC or Clang for C, CPython for Python):

- a. C programs and Python programs are both translated into machine code, which the processor executes directly.
- b. A C program is translated into machine code, which the processor executes directly; a Python program is translated into bytecode, which an interpreter executes.
- c. A C program is translated into bytecode, which an interpreter executes; a Python program is translated into machine code, which the processor executes directly.
- d. C programs and Python programs are both translated into bytecode, which an interpreter executes.

Q2. What is the decimal number whose hexadecimal representation is $\underline{2C}_{16}$?

Q3. In the following C code fragment, `uint8_t` is the type of 8-bit unsigned integers.

```
uint8_t u = 250;  
u = u + 10;
```

Circle the (only) correct answer. After the execution of this code fragment:

- a. The value of `u` is 4.
- b. The value of `u` is 255.
- c. The value of `u` is 260.
- d. `u` has no value: the program stopped with an error at the second line.

Q4. Circle the (only) correct answer.

- a. In both Python and C, the type of a variable is fixed once and for all the first time the variable is given a value.
- b. In C, the type of a variable may change during the execution of the program; in Python, it is fixed once and for all the first time the variable is given a value.
- c. In Python, the type of a variable may change during the execution of the program; in C, it is fixed once and for all when the variable is declared.
- d. In both Python and C, the type of a variable may change during the execution of the program.

Q5. `int a = 4;`
`int *p = &a;`
`int b = *p + 1;`
`*p = b * 2;`

After the execution of this C code fragment, the value of `a` is