Informatik für Mathematiker und Physiker - AS18

Exercise 2: Expressions & Integers

Handout: 25. Sep. 2018 06:00

Due: 1. Okt. 2018 23:59

Task 1: Expressions

Open Task (https://expert.ethz.ch/solve/Pv2dF3cobfnS4Ru8S)

Task

Let `a`, `b`, `c`, and `d` be variables of type `int`.

- Which of the following character sequences are valid in the sense that they are accepted by a C++ Compiler? Explain your answer.

    1. `a = b = 5`
    2. `1 = a`
    3. `a + a++`
    4. `a + b = c + d`
    5. `a = 2 b`

- For all of the expressions that you have identified as valid, decide whether these are lvalues or rvalues, and explain your decisions.

- Determine the values of the expressions that you have identified valid and explain how these values are obtained. Which of these values are unspecified and can therefore not be determined uniquely?

Task 3: Equivalent Resistance

Open Task (https://expert.ethz.ch/solve/T2f7dPSSAwqdXWgzs)

Task

Write a program `resistance.cpp` that computes the equivalent resistance of the following wiring:
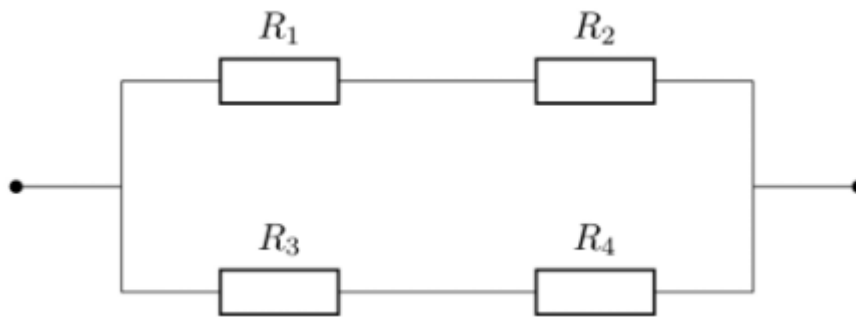
We assume that R_1, R_2, R_3, and R_4 have an integer valued resistance. After input of the four values, the program should output the result arithmetically rounded to the next integer. You may for this exercise assume that the builtin integer division rounds towards zero for all operands. **Use of floating point arithmetic is not allowed** (you must not use any `float` of `double` variable).

Remark: In order to facilitate the task, you may want to:

- conceptually divide the task into sub tasks. For example, start with computation of serial resistors R_{12} and R_{34},
- solve the task first naively using default rounding and then think about how to accomplish arithmetic rounding.

You can find formulas for computing the total resistance in this Wikipedia article (https://en.wikipedia.org/wiki/Resistor#Series_and_parallel_resistors).

Input

Four integers R_1, R_2, R_3, and R_4.

Output

An integer `R` that is equivalent to the result arithmetically rounded to the next integer.

Task 2: Representation of Integers

Open Task (https://expert.ethz.ch/solve/ewAQbph6EWM2DwzsZ)

Task

Numbers can be provided in various formats in C++. Literals prefixed with `0x` indicate hexadecimal encoding. Assume unsigned arithmetics with sufficient numbers of bits, i.e. no overflows. Convert the following hexadecimal numbers into decimal numbers:

```
1.  0x8
2.  0xc
3.  0x10c
4.  0xff
5.  0x800
6.  0x1000
7.  0x100000 (approximate)
8.  0xffffffff (approximate)
```