E1 Memory Access

The following loop

\[
\text{for } i \leftarrow 1 \text{ to } N \text{ do}\\
\quad a(i) \leftarrow (1 + s) \cdot b(i) + t \cdot c(i) \cdot d(i) + e(i)\\
\text{od}
\]

should be executed on arbitrary processor architectures. How many (machine) cycles are necessary in order to execute one full iteration if

I. a scalar processor is used which in each cycle is able to load two words or to store one word as well as to execute one multiplication or one addition,

II. a superscalar processor (MULT & ADD) is used which is able to load or to store two words per cycle?

You can assume that all scalar quantities are kept in registers during the entire program execution, write accesses (to memory) and arithmetic operations are done in parallel, and read accesses (from memory) might be executed in advance.

E2 Speedup and Parallel Efficiency

Speedup \( S \) for solving a problem on \( p \) processing elements compared to one processing element can be computed using Amdahl's law

\[
S = \frac{p}{\sigma \cdot p + (1 - \sigma)},
\]

where \( \sigma \) denotes the percentage of serial work to be done. The efficiency \( E \) of this parallel approach can be computed with \( E = S/p \).

Consider some parallel program with a percentage of serial work of 4\%. Calculate the maximum number \( p \) of processing elements, thus, the parallel efficiency \( E \) is at least 90\%.

E3 Cluster Account

In order to access the chair’s cluster (necessary for doing the practical tasks) please send an email to Christoph Ertl (christoph.ertl@tum.de) containing the following information.

- last name, first name
- TUM account (e.g. ab12cde)