Numerical Simulation and Scientific Computing I — February 3, 2022
360.242 -- Exam

Written Online Exam

General Information:

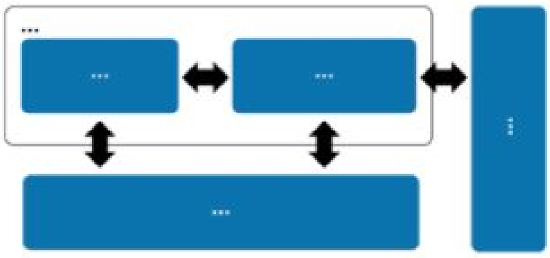
- *(**x**p) indicates the achievable points for the respective question.*
- *You must answer the questions via hand-written notes on empty sheets of paper.*
- *Show the stack of empty papers at the beginning when asked.*
- *Clearly indicate the respective question number for each answer.*
- *Add page numbers to every sheet of paper.*
- *Note the total number of submitted pages on the top of the first page.*
- *Write clearly using large hand-writing and generously use the available space.*
- *When finished notify a supervisor that you are ready to submit. You will be moved to a breakout room where you will be instructed to scan all the sheets of paper into a single PDF document. Double-check that the PDF contains all pages and is of reasonable quality and submit to: nssc@iue.tuwien.ac.at*

Inform an exam supervisor when

- *You are finished and want to scan and submit.*
- *You need more paper then initially approved by the supervisor.*
- *Something unforeseen happened.*

---

1. (2p) Consider the incomplete schematic of a certain "computer architecture" on the right. Roughly re-draw this on your hand-in and:
   a. Name this "computer architecture"
   b. Name the individual empty "..." placeholders



2. (2p) Define "Bandwidth" and "Latency" within the context of memory hierarchies on common computer architectures.

3. (2p) Describe what a "fused multiply-add" operation is (in the context of SIMD instructions).

4. (3p) Describe "Simultaneous Multithreading" **and** the importance of "Simultaneous Multithreading" for pipelines **and** provide a schematic drawing showing the relationships between: memory, caches, registers, and execution units.

5. (3p) Describe "Cache Coherence" in general **and** discuss a simplified "Cache Coherence Protocol" in words **and** by providing a schematic.

6. (4p) Consider the code snippet below, which shows a simple for-loop calculating the sum of all entries in a vector.
   a. Provide the detailed pseudocode for a four times unrolled version of this loop.
   b. Discuss the consequences on the numerical results for the final value of the sum.

```
double sum = 0.0;
for (int i = 0; i != N; ++i)
{
	sum += A[i];
}
```

Written Online Exam

7. (4p) The following C++ snippet (which tries to check if two floating point values are "almost equal") has some deficiencies. Discuss those and provide a situation (i.e., values for `a` and `b`) where these would manifest. Rewrite an improved version which resolves these deficiencies.

```
bool almost_equal(const double &a, const double &b){
  return std::fabs(a - b) < std::numeric_limits<double>::epsilon();
}
```

8. (3p) Describe the effect of and the differences between "Dirichlet", "Neumann", and "Robin" boundary conditions for the two-dimensional Laplace equation.

9. (4p) List three categories for "problem transformations" for a linear system matrix and briefly describe the effect of each transformation in words. Also comment for each category if the transformation changes the solution, and if yes, how the original solution can be obtained once the solution for the transformed system is available.

10. (2p) Consider a matrix with the following structure plot:
    a. What happens to its sparsity when applying LU decomposition without pivoting?
    b. ... and when solving it using a Krylov method?

11. (4p) Consider a system of linear equations of the form $Ax = b$ which you would like to solve for x. For each of the following characteristics of A, **suggest the best candidate method** for solving the system **and shortly explain your choice**. Suggest only one of the following 3 methods for each A: LU decomposition, GMRES, or CG.
    a. A is a 5x5 matrix with 7 zero entries.
    b. A is sparse, square, and its eigenvalues are all positive.
    c. A is a NxN matrix with $N/5$ zero entries. N is larger than 10000.
    d. A is a defective (non-diagonalizable), sparse, and square matrix.

Written Online Exam

12. (4p) Consider the following code snippet: How would you modify it in order to obtain a:
    a. Per-stream, independent random number generator?
    b. Single random number server?

```
#pragma omp parallel
{
        std::mt19937_64 seed_generator(1);
        std::uint_fast64_t seed = seed_generator();
        std::mt19937_64 thread_generator(seed);
        std::normal_distribution<double> dist(0,1);
        #pragma omp for
        for (int i = 0; i < 4; ++i)
        {
                #pragma omp critical
                std::cout << dist(thread_generator) << std::endl;
        }
}
```

13. (2p) What is the difference between a "process" and a "thread" within the context of parallel computing. Provide examples for both.

14. (2p) Explain "Amdahl's Law" in words **and** draw an explanatory "speedup" over "number of parallel units" figure ("parallel units" is meant to be a generic term for processes or threads).

15. (2p) Consider the following pseudocode for a dense matrix-matrix multiplication function of square matrices of size $n \; x \; n$:

```
matrixMatrixMultiplication(A, B) {
        for i in [0,n)
                for j in [0,n)
                        C[i][j] = 0
                        for k in [0,n)
                                C[i][j] += A[i][k]*B[k][j]
        return C
}
```

    What is its asymptotic runtime behavior in "big O" notation?
    a. $O(1)$
    b. $O(n)$
    c. $O(n \log n)$
    d. $O(n^2)$
    e. $O(n^3)$
    f. $O(n!)$

16. (3p) Why can data structures which are contiguous in memory have superior performance? In addition, give at least 2 examples of data structures which are contiguous in memory and 1 example which is not.