Executive Summary of First Class

1. The major components of any computer are its processor system, its memory system, its storage system, and its interconnect.

2. The value of interconnect is directly proportional to its bandwidth, its peak capability to move electrical signals from one place to another. Common units are GWs/s and GBs/s.

3. Although many hardware devices can "compute", high-end CPUs are amazingly uniform in that all their cores are instances of RISC 2.0 processor microarchitecture.

4. Starting in 1980, RISC computer vendors drove non-RISC computer vendors out of business. Hence the term "killer micro".

5. Domain scientists solve problems by writing, or contracting out the writing of, computer programs. Different programs require different computer capabilities in order to run fast. Since no computer can run all programs fast, computer vendors have optimized their designs to appeal to the largest user communities. This leaves many vital computer programs without an appropriate computer to run on. Some companies have pivoted to serve the machine-learning market, but this is of limited importance.

6. The fundamental way in which programs differ lies in their different memory-accessing patterns. The simplest way of characterizing this is to measure their \_arithmetic intensity\_.

7. The current fashion for highly predictable data-parallel programs has obscured the trouble we are in. These programs are easy to please because they are \_embarrassingly localizable\_.

8. Killer micros, always evolving, went through a rough patch in 2003 when chips got too hot. The single, hot, high-performance processor on a chip was replaced by multiple, cool, low-performance processors on a chip. This design strategy is called \_multicore\_. Scaling to large numbers of cores was beyond the prowess of computer vendors.

9. Most computers have plenty of compute power, and limited communication power, which makes programs with low arithmetic intensity slow.

10. Computation is limited by communication, not arithmetic. Floating-point computation is essentially free, in time and energy. In contrast, off-chip bandwidth is limited (in 2005) to a few GWs/s, and each word transferred consumes enormous energy. Feeding the FPUs with data is expensive, not the FPUs themselves. (The Japanese, in 2020, can do 32 GWs/s).

11. When an expensive, critical resource goes idle waiting for data to arrive, this is not a good thing.

12. We studied the execution of 'mul.d f0,f2,f4'.

13. We also studied a simple MIPS program with a loop.