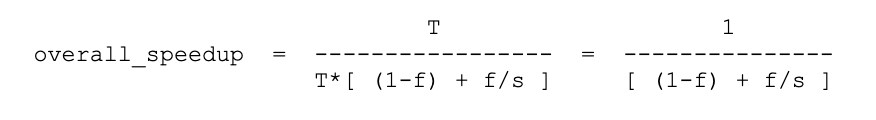
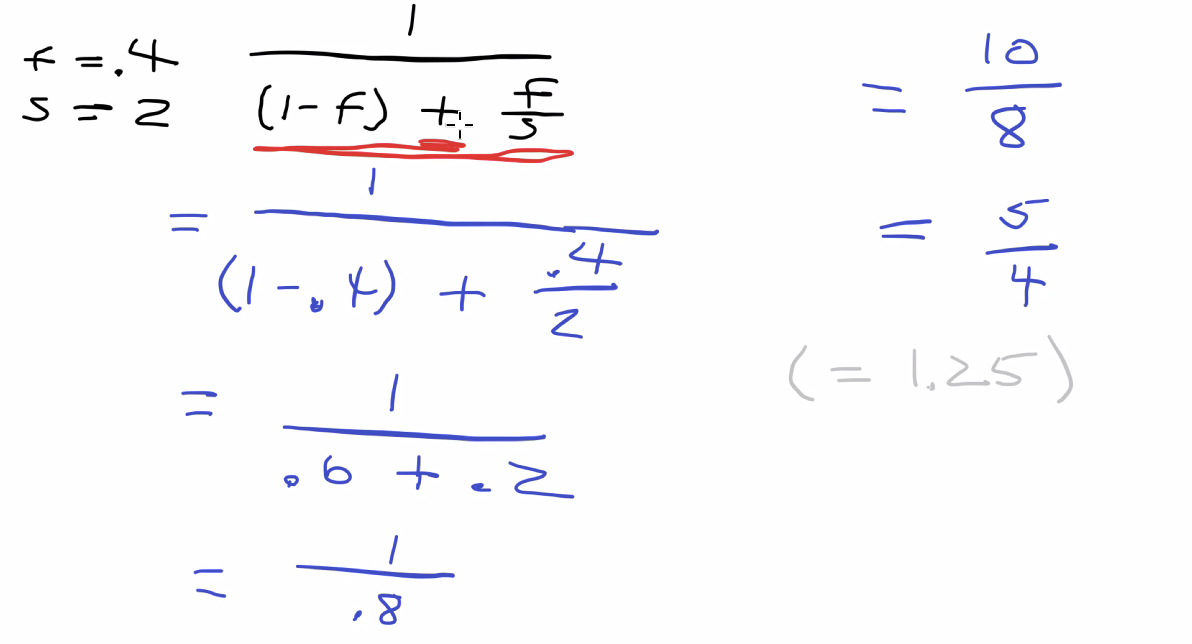
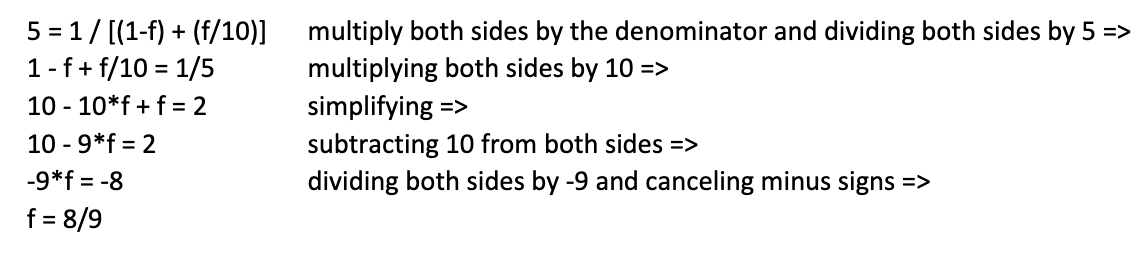
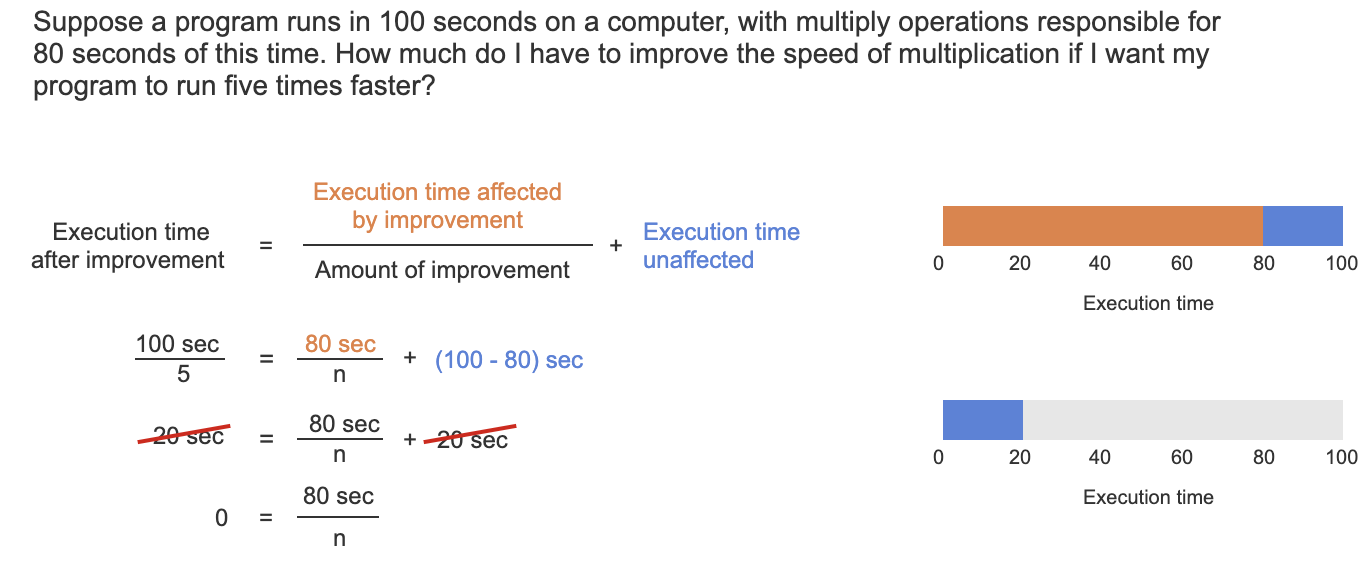
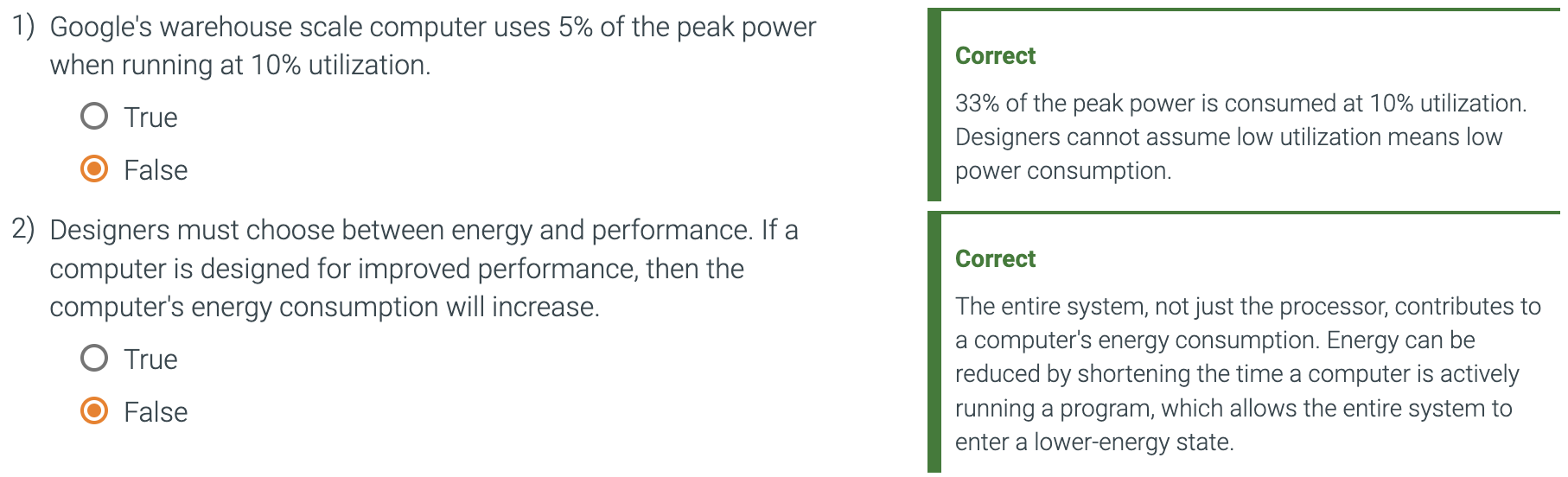
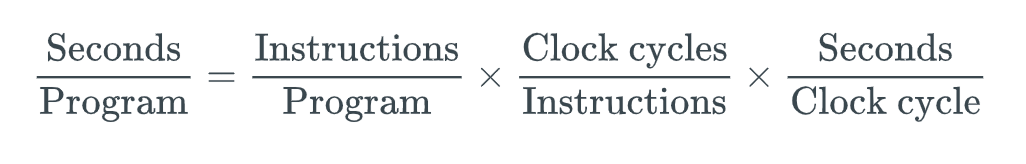
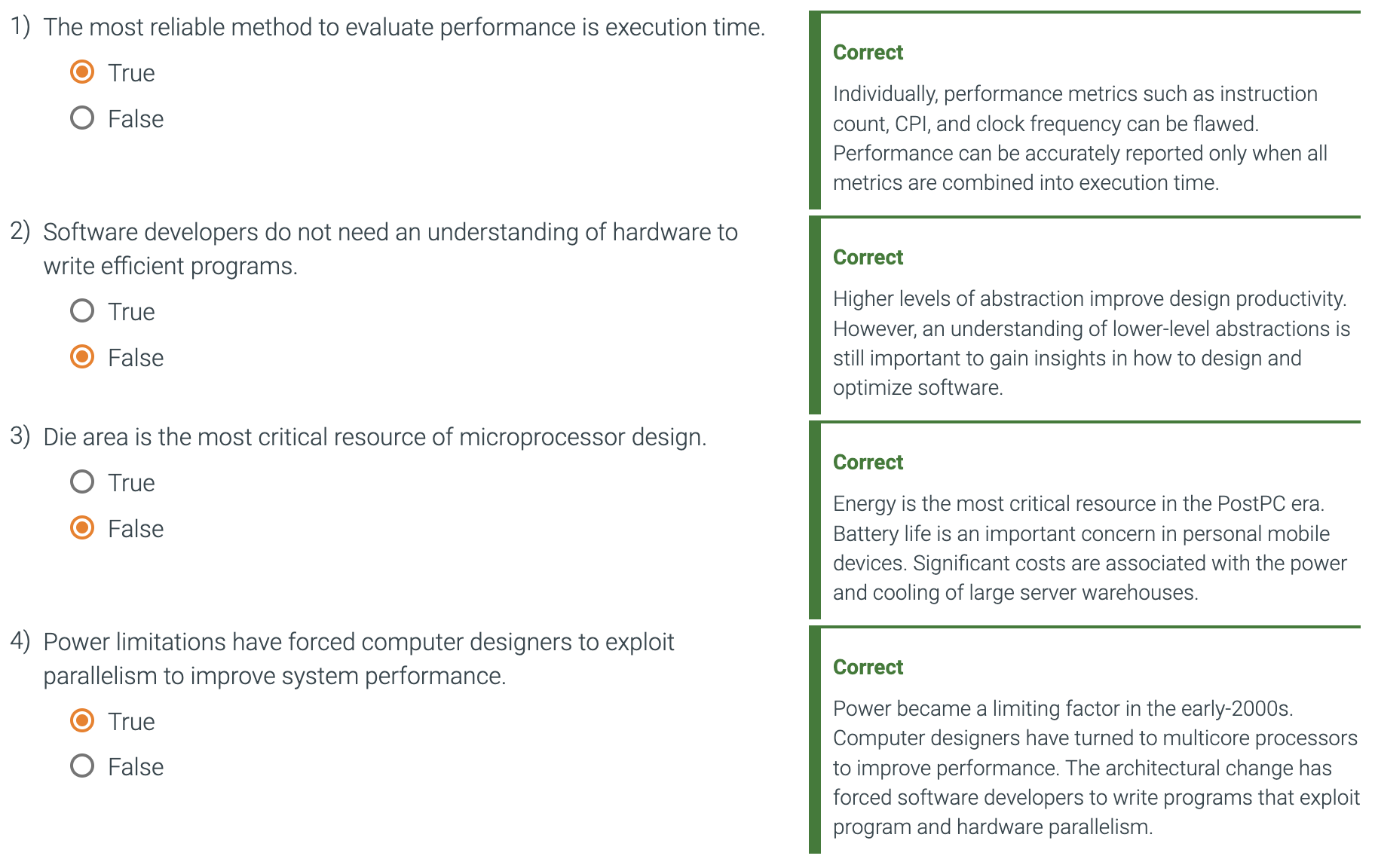
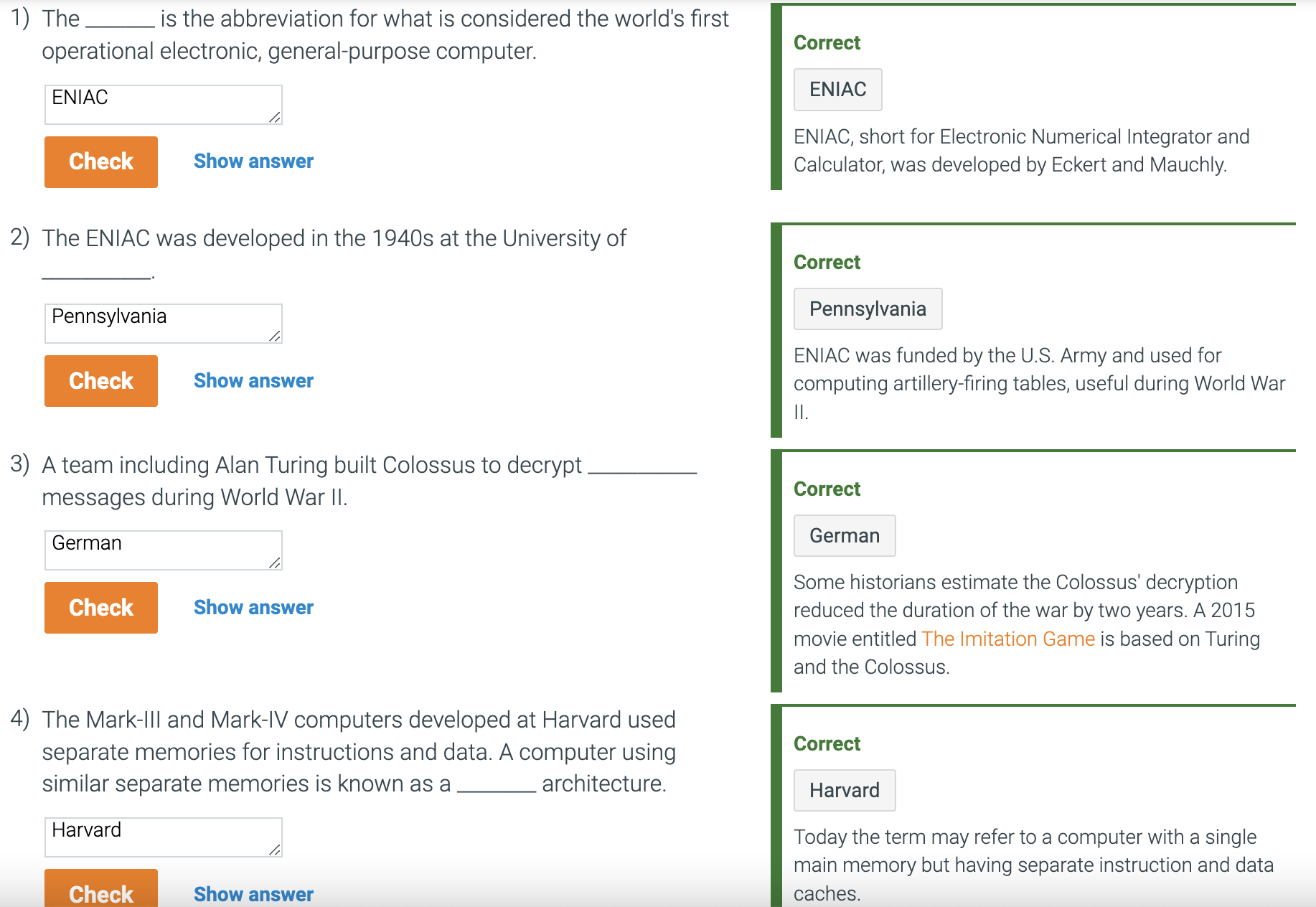
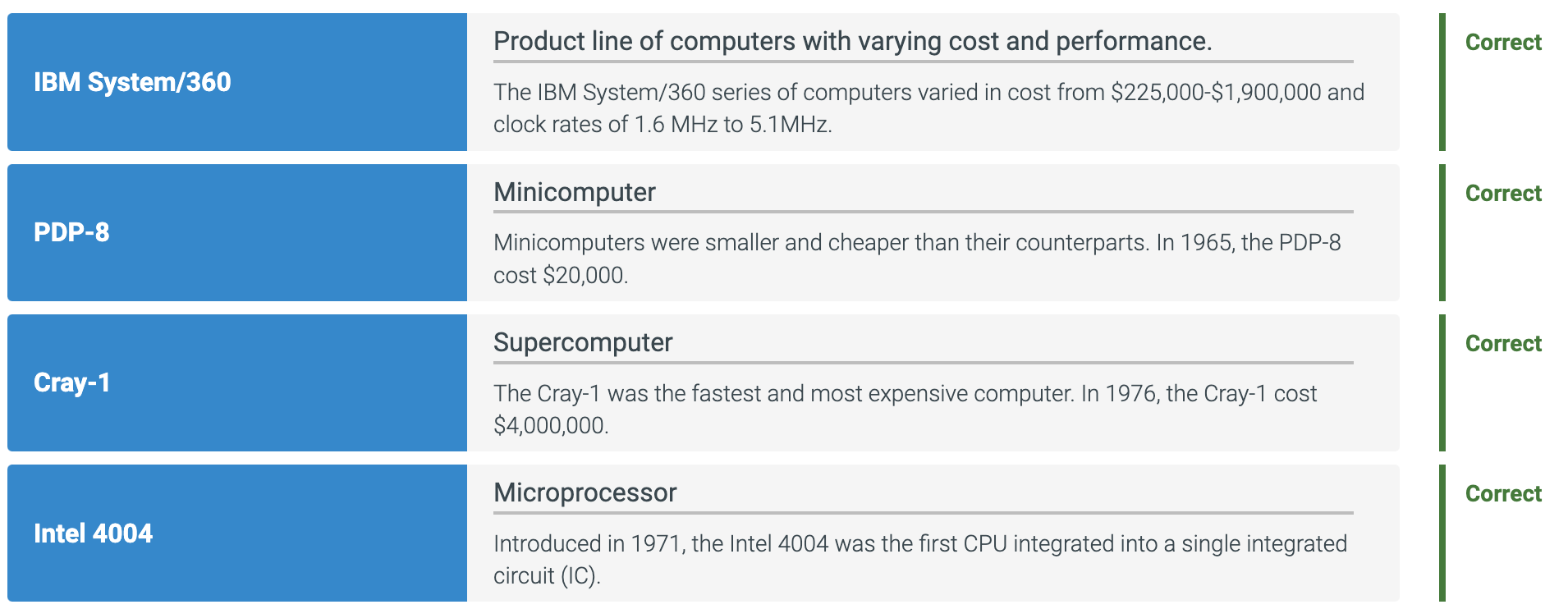
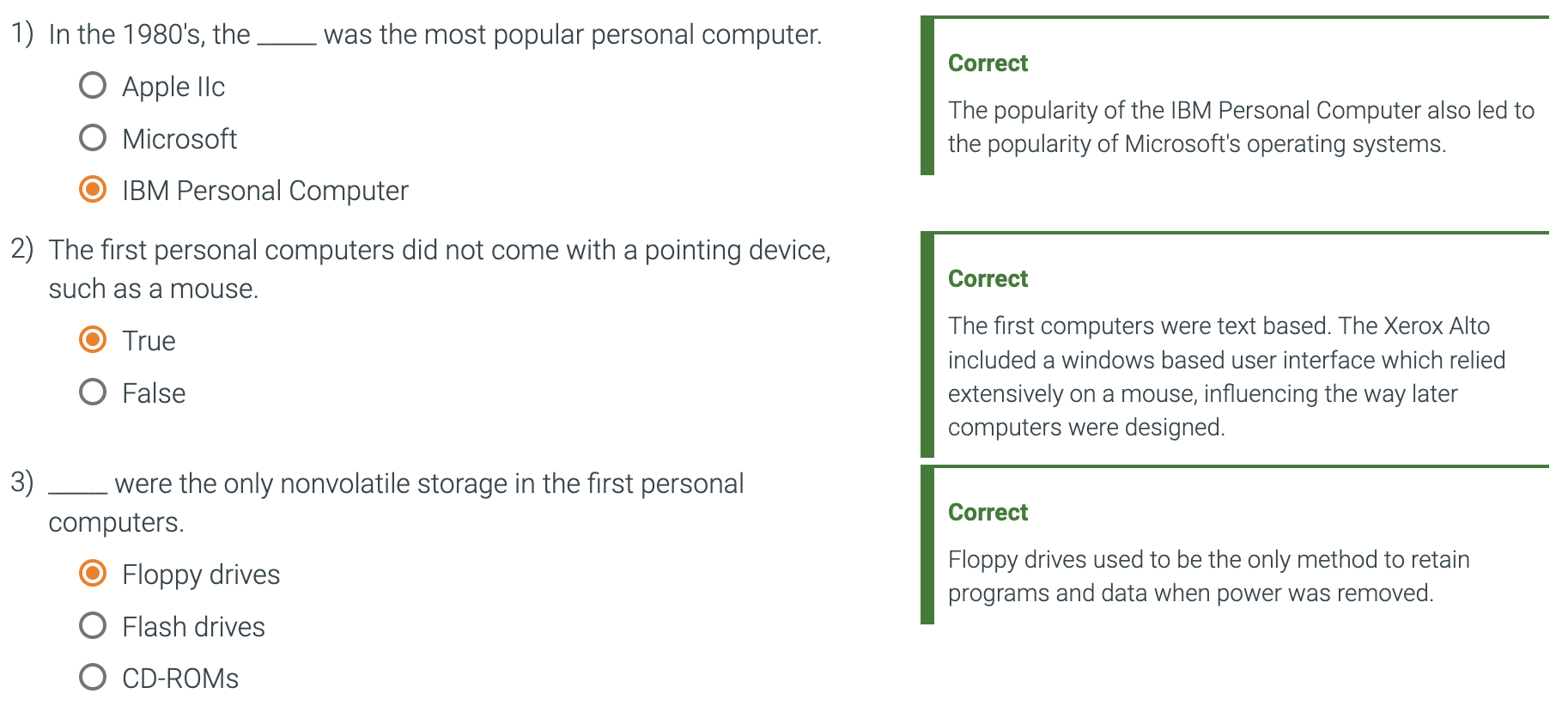
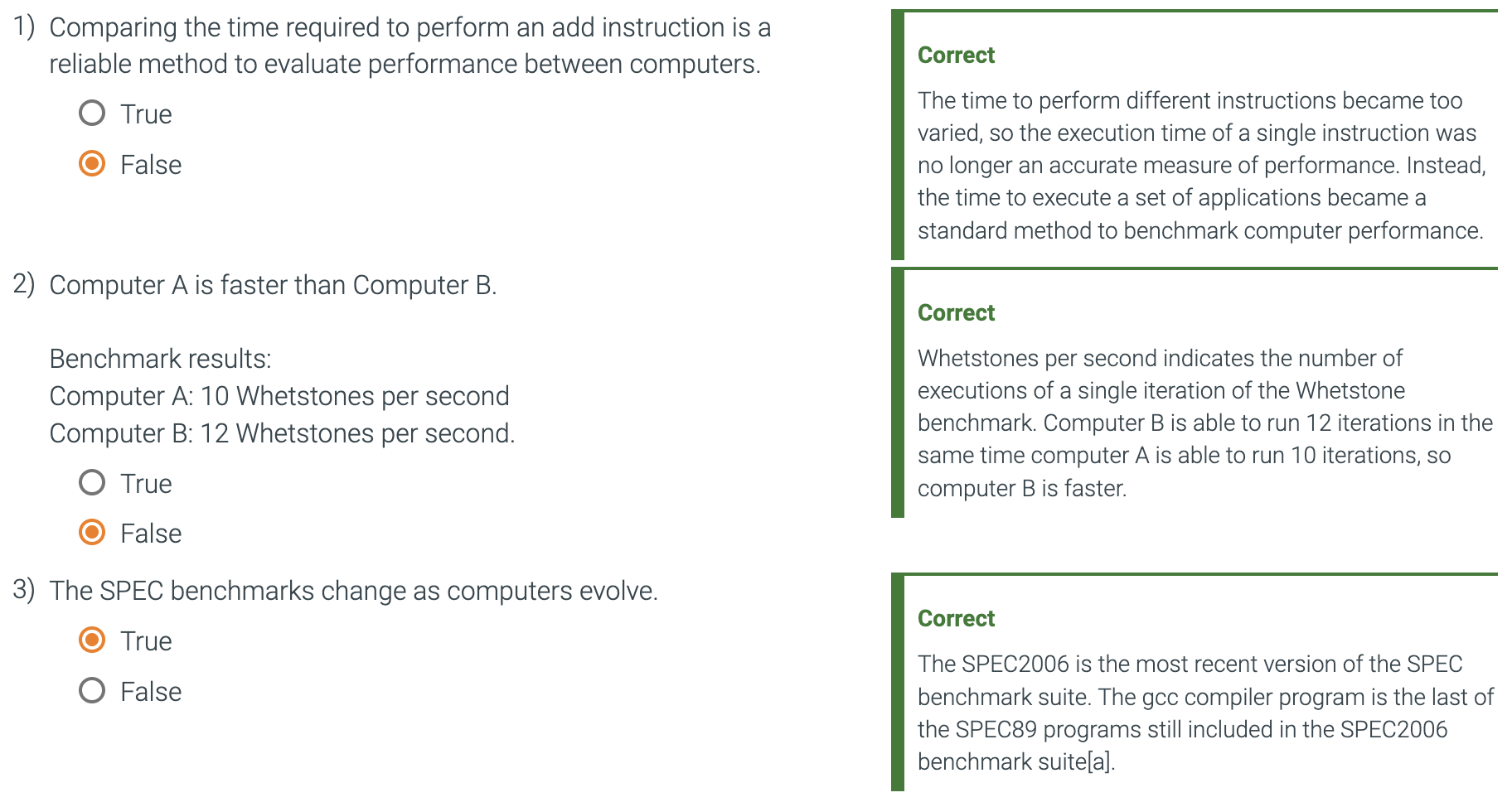
**Chapter 1.11 - 1.15 Notes**

* Lecture Notes
  + Amdahl’s Law
    - Deriving it starts from the expression
    - Overall speedup = execution time old / execution time new
    - Another way to visualize it is having multiple parallel tasks run on multiple processors for part of the execution time
      * With perfect partitioning and no communication overhead in the parallel portion
    - Its a law of diminishing returns
    - f is the program execution time
    - S is the speedup of a fraction f
  + ILLIAC IV
    - First massively parallel computer
  + The effort expended on achieving high parallel processing rates is wasted unless it is accompanied by achievements in sequential processing rates of very nearly the same magnitude.
  + Cray-1 supercomputer
    - Embodies amdahl’s wisdom.
    - Seymour cray improved the scalar performance as well as provided high performance register based vector units.
  + What is the overall speedup in program execution time if an enhancement with a speedup of 2 is available and can be used to enhance 40% of the execution time? You should use Amdahl’s Law and give your answer as a fraction.
    - Do not say that it is 25% faster than….
  + Consider a vector computer in which vector mode can provide a speedup of 10. Using Amdahl's Law, what amount of vectorization is required for an overall speedup of 5? Solve the problem algebraically and express the answer as a reduced fraction rather than a decimal or a percentage.
    - Overall speedup = 5, s = 10, find f
  + Older machines will be byte accessible.
  + Little endian addresses the least accessible byte
* Fallacies and Pitfalls 1.11
  + Pitfall
    - Expecting the improvement of one aspect of a computer to increase performance by an amount proportional to the size of the improvement.
  + Amdahl’s Law
    - A rule stating that the performance enhancement possible with a given improvement is limited by the amount that the improved feature is used.
    - It is a quantitative version of the law of diminishing returns.
    - We use it to estimate performance improvements when we know the time consumed for some function and its potential speedup.
    - Amdahl’s law, with the cpu performance equation, is a handy tool for evaluating potential enhancements.
    - Also used to argue for practical limits to the number of parallel processors.
    - For above, the multiply operations must run in 0 seconds to run the program 5 times faster.
    - 30 / 2 + (100 - 30)
      * 15 + 70
      * 85 seconds
    - If some aspect of a computer accounts for 50% of program execution time, what is the limit on how many times faster programs can run if engineers focus on improving that aspect?
      * 2 times
  + Fallacy: computer at low utilization use little power
    - Power efficiency matters at low utilizations because server workloads vary.
    - Even given five years to learn how to run the SPECpower benchmark well, the specially configured computer with the best results in 2020 still uses 33% of the peak power at 10% of the load.
    - If future servers used, say, 10% of peak power at 10% workload, we could reduce the electricity bill of datacenters and become good corporate citizens in an era of increasing concern about CO2 emissions.
  + Fallacy: Designing for performance and designing for energy efficiency are unrelated goals.
  + Pitfall: Using a subset of the performance equation as a performance metric.
    - One alternative to time is MIPS
    - MIPS
      * Million instructions per second
      * A measurement of program execution speed based on the number of millions of instructions.
      * MIPS is computed as the instruction count divided by the product of the execution time and 10^6
    - Three problems with MIPS
      * MIPS specified the instruction execution rate but does not take into account the capabilities of the instructions.
      * MIPS varies between programs on the same computer, thus a computer cannot have a single MIPS rating.
      * If a new program executes more instructions but each instruction is faster, MIPS can vary independently from performance.
* Concluding Remarks 1.12
  + Hardware and software designers construct computer systems in hierarchical layers, with each lower layer hiding details from the level above.
    - This is abstraction.
  + The most important example of abstraction is:
    - The interface between hardware and low-level software called the instruction set architecture.
  + Execution time is the only valid and unimpeachable measure of performance.
  + The key hardware technology for modern processors is silicon.
  + Two of the key ideas are exploiting parallelism in the program, typically today via multiple processors, and exploiting locality of accesses to a memory hierarchy, typically via caches.
  + Energy efficiency has replaced die area as the most critical resource of microprocessor design.
    - Conserving power while trying to increase performance has forced the hardware industry to switch to multi core microprocessors, thereby forcing the software industry to switch to programming parallel hardware.
    - Parallelism is now required for performance.
* Historical Perspective and Further Reading 1.13
  + J. Presper Eckert and John Mauchly at the Moore School of the University of Pennsylvania build the first operational electric general purpose computer called the **ENIAC** (electronic numerical integrator and calculator)
    - Funded by the United States Army
    - Operational during WWII, not publicly disclosed until 1946
    - Used for computing artillery firing tables
    - U-shaped
    - 80 feet long, 8.5 feet high, several feet wide
    - Each of the 20 10-digit registers was 2 feet long
    - Used 18,000 vacuum tubes
    - 8 orders of magnitude slower than modern machines, performing 1900 additions per second
    - Programming was done manually by plugging cables and setting switches, data was entered on punched cards.
      * For typical calculations, programming could take half an hour to a whole day.
    - Limited by a small amount of storage and tedious programming.
  + John von Neumann proposed a stored-program computer called **EDVAC** (Electronic Discrete Variable Automatic Computer)
    - Herman Goldstine distributed the memo but put von Neumann’s name on it.
    - Eckert and Mauchly worked on the machine but were not credited.
    - Eckert and Machly left the project in 1947 and this delayed completion until 1952.
  + Maurice Wilkes decided to build a stored-program computer named **EDSAC** (Electronic Delay Storage Automatic Calculator)
    - Became operational in 1949
    - World’s first full scale operational stored-program computer
      * A small prototype called the **Mark-I** (University of Manchester, 1948) is the first operational stored-program machine
  + IAS machine was built by Julian Bigelow
    - Has a total of 1024 40-bit words, 10 times faster than ENIAC
    - Machine was inspired by the memo written by von Neumann memo
  + John Atanasoff
    - Built a machine in the 40s at Iowa State
    - Special purpose computer that wasn't completely operational
  + Konrad Zuse built a special purpose machine in late 30s/early 40s in Germany
    - Government didn’t fund it because of the war
  + Alan Turing built the Colossus in 1943 with a team at Bletchley Park
  + Howard Aiken built the Mark-I at Harvard
    - Followed this machine with a relay machine Mark-II, and a pair of vacuum tube machines, Mark-III and Mark-IV
    - Mark-III and Mark-IV had separate memories for instructions and data
    - Regarded as reactionary by advocates of stored program computers
  + Harvard Architecture
    - Coined to scribe machines with separate memories
    - Used today in a different sense, to describe machines with a single main memory but separate caches for instructions and data.
  + Whirlwind Project
    - Begun at MIT in 1947
    - Aimed at applications in real-time radar signal processing
    - Most important innovation was magnetic-core memory
    - Has 2048 16-bit words of magnetic core
    - Magnetic cores served as the main memory for 30 years
  + In December 1947, Eckert and Mauchly formed Eckert-Mauchly Computer Corporation.
    - Their first machine, the **BINAC**, was built for Northrop and was shown in August 1949
    - The firm was eventually acquired by Remington-Rand where they built the **UNIVAC (Universal Automatic computer)**
      * Designed to be sold as a general purpose computer
      * Correctly predicted the 1952 presidential election, withheld from broadcasting networks though.
      * First delivered in June 1951, **UNIVAC I** sold for about $1 Million and was the first successful commercial computer.
  + The IBM 701, first IBM computer, was shipped in 1952, 19 units were sold.
  + In 1964, after investing $5 Billion, IBM announced the **System/360.**
    - A family of computers developed to address a wide range of computing needs
    - Four models that varied in cost and performance by a factor of almost 10, it grows to a factor of 25 if we include models 20 and 30.
      * Model 40, 1.6 MHz, 32 KB–256 KB, $225,000
      * Model 50, 2.0MHz, 128KB–256KB, $550,000
      * Model 65, 5.0MHz, 256KB–1MB, $1,200,000
      * Model 75, 5.1MHz, 256KB–1MB, $1,900,000
  + In 1965, **Digital Equipment Corporation (DEC)** unveiled the **PDP-8.**
    - The world’s first minicomputer
    - Small machine that was under $20,000.
    - Minicomputers were the forerunners of microprocessors
      * Intel invited the first microprocessor in 1971, the Intel **4004**
  + In 1963, the first supercomputer was announced
    - Extremely fast computers targeted to perform a large number of computations typically needed by scientific applications.
  + Seymour Cray led the design of the Control Data Corporation CDC 6600 in Minnesota and is the ‘father of supercomputing’.
    - Formed Cray Research INC in Wisconsin
    - In 1976, he announced the Cray-1.
      * The fastest in the world, the most expensive, and the computer with the best cost/performance for scientific programs.
      * Sales only a little over 80 supercomputers
  + Apple llc set standards for low cost, high volume, and high reliability.
    - Between 1977 and 1993, Apple produced about six million of these.
  + The IBM Personal Computer, announced in 1981, became the most popular microprocessor, with Microsoft being the most popular OS.
  + The first personal computers were quite simple, with little or no graphics capability, no pointing devices, and primitive operating systems compared to those of today.
  + Xerox Alto
    - Created as an experimental prototype of a future computer.
    - Inspired many architectural and software concepts that characterize modern desktops
    - Hundreds were made and many donated to universities
    - Technologies that it incorporated were:
      * A bit-mapped graphics display integrated with a computer
      * A mouse
      * A local area network (LAN) and was a precursor to the internet
      * A user interface based on windows featuring a what you see is what you get editor and interactive drawing programs
      * Both file servers and print servers were developed and interfaced via the local area network.
        + connections between LAN and wide are ARPAnet produced the versions of internet-stype networking
  + Original measure of performance was the time required to perform an individual operation, such as addition.
    - To consider differences like timing, etc, an instruction mis was calculated by measuring the relative frequency of instructions in a computer across many programs.
    - Multiplying the time each instruction took by its weight in the mix gave the user the average instruction time.
  + The Whetstone was a synthetic program created by measuring scientific programs written in Algol-60
    - Conveyed to Frotran and was widely used to characterize scientific program performance.
    - Typically quoted in Whetstones per second (number of executions of a single iteration of the Whetstone benchmark)
    - Dhrystone is another benchmark that is still used in some embedded computing circles.
  + Around the same time of Whetstone, kernel benchmarks became popular
    - Kernels are small, time-intensive pieces from real programs that are extracted and then used as benchmarks
    - Developed for benchmarking high-end computers, especially supercomputers
    - Livermore Loops and Linkpack are best examples.
      * The Livermore Loops consist of a series of 21 small loop fragments.
      * Linpack consists of a portion of a linear algebra subroutine package
    - Kernels often overstate the performance on real applications
    - Best used to isolate the performance of individual features of a computer and explain reasons for differences in the performance of real programs
  + An important advance in performance evaluation was the formation of the System Performance Evaluation Cooperative (SPEC) group in 1988.
    - SPEC comprises representatives of many computer companies—the founders being Apollo/Hewlett-Packard, DEC, MIPS, and Sun.
    - In 1991, a throughput measure was added, based on running multiple versions of the benchmark.
    - Other system benchmarks that include OS-intensive and I/O-intensive activities have also been added.
    - SPEC98 contained six floating-point benchmarks, but only four integer benchmarks
      * Calculating a single summary measurement using the geometric mean of execution times normalized to a VAX-11/780 meant that this measure favored computers with strong floating-point performance.
    - SPEC92 was introduced and incorporated additional benchmarks, dropped matrix300, and provided separate means for integer and floating-point programs (SPEC INT and SPECFP)
    - The SPECbase measure, which disallows program-specific optimization flags, was added to provide users with a performance measurement that would more closely match what they might experience on their own programs.
    - SPECFP showed the largest increase in numbers verses the base SPECFP measurement
    - The sole survivor from SPEC89 is the gcc compiler
  + Embedded processors have been around for a long time: The first minicomputers and first microprocessors were originally developed for controlling functions in a lab or industrial application.
    - Embedded processors grew in popularity due to their needs from video games to cell phones, etc.
    - The Embedded Microprocessor Benchmark Consortium (EEMBC) was created in 1997
      * A second generation of these benchmarks was announced in 2007.
    - In 2019, Embench was developed with the goal of replacing widespread use of synthetic programs like Dryhstone for embedded computing benchmarking.