1. Investigate the specification of your laptop, in terms of processor type, number of processors, processor speed, amount of memory, amount of disk space, etc.
   1. Processor Type: Intel® Core™ i7-2677M CPU @ 1.80GHz
   2. Number of Processors: One
   3. Processor Speed: 1.80GHz
   4. RAM: 4.00GB
   5. System Type: 64-bit operating system (Windows 10), x64 based processor
   6. Disk Space: 119GB
   7. Brand: ASUS
2. Investigate the specification of your mobile phone (and your tablet, if you have one), in terms of processor type, number of processors, processor speed, amount of memory, amount of "disk" space, etc.
   1. Brand: Apple
   2. Model: iPhone 6 Plus
   3. Processor Type:
      1. A8 chip with 64-bit architecture
      2. M8 motion coprocessor
   4. Number of Processors:
      1. 2 (see b. for details)
   5. Processor Speed:
      1. A8 chip: 1.4GHz
      2. M8 motion coprocessor: 150MHz
   6. Amount of Memory:
      1. RAM: 1 GB LPDDR3 RAM
      2. Disk Space: 128GB
   7. More facts: <https://support.apple.com/kb/SP706?locale=en_US>
3. What is the “von Neumann Bottleneck”?
   1. The “von Neumann Bottleneck” (vNB), which occurs in von Neumann computer architecture[[1]](#footnote-1), is latency[[2]](#footnote-2), the delay between the input and the consequent desired output of a system, caused by the differences between Central Processing Unit (CPU) speed and the rate of data transfer between memory and the CPU. Intuitively, the CPU is faster than the transfer of information from memory to the CPU.

A consequence of the vNB is that information transfer between memory and the CPU, called throughput[[3]](#footnote-3), is a non-trivial limit to advances in computing speed: while CPU speed may increase rapidly, throughput between the CPU and memory limits overall processing power as the CPU will have moments of inactivity while it waits for information from memory. Colloquially, overall computing speed is only as quick as its slowest component and throughput between memory and the CPU currently lags CPU speed, which is the vNB.

1. How many bytes of memory can be addressed in a 4-bit architecture computer?
   1. 1 byte = 8 bits = 23 bits
   2. A 4-bit architecture can store 24 bits, so 24 bits \* (1 bytes / 23 bits) = 21 bytes
2. How many bytes of memory can be addressed in an 8-bit architecture computer?
   1. 1 byte = 8 bits = 23 bits
   2. An 8-bit architecture can store 28 bits, so 28 bits \* (1 bytes / 23 bits) = 25 bytes
3. How many bytes of memory can be addressed in a 16-bit architecture computer?
   1. 1 byte = 8 bits = 23 bits
   2. A 16-bit architecture can store 216 bits, so 216 bits \* (1 bytes / 23 bits) = 213 bytes
4. How many bytes of memory can be addressed in a 32-bit architecture computer?
   1. 1 byte = 8 bits = 23 bits
   2. A 32-bit architecture can store 232 bits, so 232 bits \* (1 bytes / 23 bits) = 229 bytes
5. How many bytes of memory can be addressed in a 64-bit architecture computer?
   1. 1 byte = 8 bits = 23 bits
   2. A 64-bit architecture can store 264 bits, so 264 bits \* (1 bytes / 23 bits) = 261 bytes
6. Perform the following additions in Base 10:
   1. 123410 + 123410 = 246810
   2. 123410 + 278910 = 402310
   3. 88810 + 88810 = 177610
7. Perform the following additions in Base 2:
   1. 10102 + 01012 = 11112
   2. 10102 + 01112 = 100012
   3. 1112 + 12 = 10002
8. What is the relationship between Base 2 and Base 8?
9. What is the relationship between Base 2 and Base 16?

1. See here for an overview: <https://en.wikipedia.org/wiki/Von_Neumann_architecture> [↑](#footnote-ref-1)
2. <https://en.wikipedia.org/wiki/Latency_(engineering)> [↑](#footnote-ref-2)
3. <https://en.wikipedia.org/wiki/Von_Neumann_architecture> [↑](#footnote-ref-3)