ASSIGNMENT

…

High Performance Computing
IS5103

21CIS0093
MMDCK THILAKARATHNA

Question 1
What does HPC stand for and what is its primary purpose?

HPC = High Performance Computing. Its primary role is to provide very high computing capability (servers, clusters, supercomputers + software, storage, networking) to address computationally, memory- or data-limited scientific, engineering and analytical problems (e.g., climate/weather, CFD, astrophysics, large-scale simulations).

Question 2
According to the TOP500 list, what operating system is used by all supercomputers?

Linux (Linux variants are used across the TOP500).

Question 3
What does FLOPS measure in HPC systems?

FLOPS = Floating Point Operations Per Second. Used to quantify the system's floating-point arithmetic operation rate (standard HPC performance indicator).

Question 4
Name the current #1 supercomputer and its location.
Frontier located at Oak Ridge National Laboratory (ORNL), USA

Question 5
List the three main elements of the HPC ecosystem.

Hardware, Software, and Problems (workloads/data & people)

Question 6
Explain the difference between theoretical peak performance and sustained performance in HPC systems.
Theoretical peak performance (Rpeak) — derived from hardware specifications:
FLOPS_peak = clock_rate × FP_ops_per_cycle × number_of_cores (total across sockets). It is an upper limit best-case given each cycle each core is doing the maximum FP ops.

Sustained performance (Rmax) — measured by running real codes or benchmarks (e.g., HPL/Linpack). It is total_FP_operations_executed / time_taken and it is real performance with memory, communication, and inefficiencies accounted for. TOP500 uses the HPL-measured sustained performance.

Question 7
The lecture mentions climate modeling over the Mediterranean Sea. If a 200km resolution model requires ~1GB of RAM:

a) **Calculate the RAM needed for a 2km resolution model**

   Linear resolution factor = 200 / 2 = 100
   For a 3-D grid, number of cells scales with factor= 100^3 = 1,000,000
   So, RAM ≈ 1 GB × 1,000,000 = 1,000,000 GB = 1,000 TB

b) **Explain why higher resolution requires more resources**
   - Higher spatial resolution → significantly more grid points (scales as the cube for 3D) → additional memory to keep state variables at each grid point.

   - Higher compute per time step (higher number of grid points to advance) → increased runtimes or extra CPU/GPU resources.

   - Increased I/O (more data to transmit/store), and more demanding communication needs in distributed runs (network bandwidth / latency).

Question 8

For the HPL (High Performance Linpack) benchmark:

a) What mathematical problem does it solve?

Solves a dense system of linear equations. Ax=b for a randomly generated dense matrix and prints time and FP rate.

b) What is the formula for calculating floating-point operations?

FLOPs=$32n^3+2n^2$(where n is matrix size)

c) If a system takes 100 seconds to solve a problem with n=50,000, calculate the sustained FLOPS performance.

If time = 100 s, n = 50,000 → sustained FLOPS:

Compute operations:

- $n^3=(50,000)^3=1.25\times10^{14}$
- $2/3n^3=2/3\times1.25\times10^{14}=8.333333\times10^{13}$
- $2n^2=2\times(50,000)^2=2\times2.5\times10^9=5.0\times10^9$

- Total FLOPs ≈ 8.3338333×10^{13}
- Sustained FLOPS = total / time = $8.3338\times10^{13}/100=8.3338\times10^{11}$

GFLOPS ≈ 0.833 TFLOPS

Question 9

Analyze the concept of "Performance vs Productivity" in HPC:

a) Provide the productivity formula

Productivity=application programming effort/application performane

b) Give a detailed scenario where performance improvement isn't worthwhile

Example: Suppose a baseline code runs in 1000 s. An optimization team is able to reduce runtime to 500 s (2× speedup) but in a 6-month (≈ 1000 developer-hours) undertaking, based on major code refactoring and sophisticated vendor-tuned libraries that increase the overhead of maintenance. If only occasional runs are necessary (e.g., every month) or the expense of person-hours exceeds the operation savings (or the scientific turnaround time advantage is small), then the ROI is negative — 2× performance is not worth the development cost. Also if the enhanced code is brittle (hard to port, maintain, reproduce), productivity (scientist time, reproducibility) drops. The lecture uses an analogous example (6 months needed for 2× speedup) as well.

c) **Explain why some suggest HPC should mean "High Productivity Computing"**

Raw computing performance is not enough — scientific productivity depends on how quickly and reliably researchers can produce, run, and interpret results. Productivity is concerned with developer/user time, portability, maintainability, and end-to-end time-to-science, not just FLOPS. Streamlining systems to program and utilize can increase scientific throughput more than incremental increases in peak performance.

Question 10

Discuss the evolution from traditional HPC to HPDA (High Performance Data Analytics):

a) **Define HPDA and contrast it with traditional HPC**

HPDA (High Performance Data Analytics): running data-heavy analytics workloads (large data processing, statistical analysis, streaming analytics, large-scale ML/AI inference and training) on HPC-class infrastructure and parallel data frameworks.

Contrast: Classic HPC focuses on simulation/numerical modeling (compute-bound, regular dense or sparse computational kernels). HPDA is more about data (I/O, streams, out-of-core access patterns, ML pipelines) and follows a different software stack (data platforms, ML libraries), but more and more on common converged hardware.

b) How has AI influenced this evolution?
AI (training and inference) has made necessary: enormous parallel matrix/tensor compute demand, fast hardware (GPUs/TPUs), fast libraries (cuDNN, TensorFlow/PyTorch), and coupled workflows where simulation output is fed as input to ML models and ML informs simulation decisions. This confluence prompted HPC centers to offer AI/HPDA stacks (GPU clusters, mixed storage/IO modes, software suites) see the course lecture on HPC → HPDA & AI.

Question 11
Analyze the "Inference Spiral of System Science" and its implications:

a) Describe what the spiral represents

The Inference Spiral shows how science is a cycle: models get bigger, generate new predictions/insights, eat up new data, which prompts better or bigger models the cycle continues. In every iteration, data amount and model size tend to grow, creating a spiral of increasing computational and data needs.

b) Justify why it demands ever-increasing computational power

Because every cycle adds model resolution/complexity or uses more data (finer grids, higher fidelity, more parameters, larger training sets), which costs more computational operations, memory, storage, and communications thus increasingly greater amounts of compute resources are required to run the next cycle of experiments/anlysis within reasonable time.

Question 12

FLOPS Calculations A CPU has the following specifications:

- **Clock frequency: 3.2 GHz**
- **8 cores per processor**
- **Each core can perform 2 floating-point operations per clock cycle**
- **The system has 4 processors**

Calculate:

a) Theoretical peak performance in FLOPS

Theoretical peak performance (FLOPS):

Peak=3.2×10^9 cycles/s×2 FP/cycle×8 cores×4 sockets

Stepwise:

- 3.2e9 × 2 = 6.4e
- ×8 = 51.2e9
- ×4 = 204.8e9 FLOPS

204.8×10^9 FLOPS = 204.8 GFLOPS.

b) Convert to appropriate prefix notation

Appropriate prefix: 204.8 GFLOPS (≈0.2048 TFLOPS)

c) If sustained performance is 65% of peak, what is the actual FLOPS?

If sustained = 65% of peak:

Sustained = 0.65 × 204.8e9 = 133.12×10^9 FLOPS = 133.12 GFLOPS

Question 13

HPL Benchmark Calculation A supercomputer runs the HPL benchmark with a matrix size n = 100,000. The benchmark completes in 450 seconds.

Calculate:

a) Total floating-point operations using HPL formula

total FLOPs:

- $n3=(10^5)^3=10^{15}$
- $2/3n^3=2/3\times10^{15}=6.6666667\times10^{14}$
- $2n^2=2\times10^{10}=$

Total ≈ 6.6668667×10^{14} FLOPs (keeping small term included).

b) Sustained performance in FLOPS

total / time = $6.6669\times10^{14}/450\approx1.4815$FLOPS.

c) Convert to petaFLOPS

1.4815×10^{12} FLOPS = 0.0014815 PFLOPS

Question 14

Data Transfer Calculations A scientific application needs to transfer data between nodes:

- **File size: 2.5 TB**
- **Network bandwidth: 100 Gbps**
- **Network efficiency: 80%**

Calculate:

a) Effective bandwidth

Effective bandwidth = 100 Gbps × 0.80 = 80 Gbps.

b) Transfer time in seconds

convert file to bits: 2.5 TB → assume decimal TB = 2.5×10^{12} bytes × 8 = 2.0×10^{13} bits

Time = bits / bandwidth = 2.0×10^{13} bits / 80×10^9

$=2.0\times10^{13}/8.0\times10^{10}=250$ s.

c) Transfer time in minutes

250/ 60 = 4.1667 minutes

c) **If 10 nodes transfer simultaneously, what's the per-node bandwidth**

If 10 nodes transfer simultaneously and bandwidth is shared equally: per-node effective bandwidth = 80 Gbps / 10 = 8 Gbps per node (so each node effectively gets 8 Gbps). If you quote raw link share before efficiency, raw per-node = 100/10 = 10 Gbps (but with 80% efficiency the useful per-node rate is 8 Gbps)

Question 15

Power Consumption Analysis Frontier supercomputer specifications:

- **Peak performance: 1,679.82 PFLOPS**
- **Power consumption: 22,703 kW**
- **Operates 70% of the time annually**
- **Electricity cost: $0.08 per kWh**

Calculate:

a) **Power efficiency in FLOPS/Watt**

Power efficiency (FLOPS/W): use peak FLOPS / power (in Watts). We can express as GFLOPS/W or FLOPS/W.

Power in Watts = 22,703 kW × 1000 = 22,703,000 W

FLOPS/W = 1.67982×10^{18}/ 2.2703×10^{7} = 7.399×1010 FLOPS/W.

b) **Annual energy consumption in kWh**

Hours per year = 24×365 = 8,760 h

Operates 70% → 8,760 × 0.70 = 6,132 hours.

Energy (kWh) = Power (kW) × hours = 22,703 kW × 6,132h =139,214,796 kWh

c) **Annual electricity cost**

Annual electricity cost = energy × $0.08/kWh = 139,214,796 × 0.08 = $11,137,183.68 ≈ $11.14 million per year.