High Performance Computing (HPC) MCQs [set-2]

26. NVIDIA CUDA Warp is made up of how many threads?

A. 512

B. 1024

C. 312

D. 32

Answer: D

27. Out-of-order instructions is not possible on GPUs.

A. true

B. false

C. --

D. --

Answer: B

28. CUDA supports programming in ....

A. c or c++ only

B. java, python, and more

C. c, c++, third party wrappers for java, python, and more

D. pascal

Answer: C

29. FADD, FMAD, FMIN, FMAX are ----- supported by Scalar Processors of NVIDIA GPU.

A. 32-bit ieee floating point instructions

B. 32-bit integer instructions

C. both

D. none of the above

Answer: A

30. Each streaming multiprocessor (SM) of CUDA herdware has ------ scalar processors (SP).

A. 1024

B. 128

C. 512

D. 8

Answer: D

31. Each NVIDIA GPU has ------ Streaming Multiprocessors

A. 8

B. 1024

C. 512

D. 16

Answer: D

32. CUDA provides ------- warp and thread scheduling. Also, the overhead of thread creation is on the order of ----.

A. “programming-overhead”, 2 clock

B. “zero-overhead”, 1 clock

C. 64, 2 clock

D. 32, 1 clock

Answer: B

33. Each warp of GPU receives a single instruction and “broadcasts” it to all of its threads. It is a ---- operation.

A. simd (single instruction multiple data)

B. simt (single instruction multiple thread)

C. sisd (single instruction single data)

D. sist (single instruction single thread)

Answer: B

34. Limitations of CUDA Kernel

A. recursion, call stack, static variable declaration

B. no recursion, no call stack, no static variable declarations

C. recursion, no call stack, static variable declaration

D. no recursion, call stack, no static variable declarations

Answer: B

35. What is Unified Virtual Machine

A. it is a technique that allow both cpu and gpu to read from single virtual machine, simultaneously.

B. it is a technique for managing separate host and device memory spaces.

C. it is a technique for executing device code on host and host code on device.

D. it is a technique for executing general purpose programs on device instead of host.

Answer: A

36. _____ became the first language specifically designed by a GPU Company to facilitate general purpose computing on ____.

A. python, gpus.

B. c, cpus.

C. cuda c, gpus.

D. java, cpus.

Answer: C

37. The CUDA architecture consists of --------- for parallel computing kernels and functions.

A. risc instruction set architecture

B. cisc instruction set architecture

C. zisc instruction set architecture

D. ptx instruction set architecture

Answer: D

38. CUDA stands for --------, designed by NVIDIA.

A. common union discrete architecture

B. complex unidentified device architecture

C. compute unified device architecture

D. complex unstructured distributed architecture

Answer: C

39. The host processor spawns multithread tasks (or kernels as they are known in CUDA) onto the GPU device. State true or false.

A. true

B. false

C. ---

D. ---

Answer: A

40. The NVIDIA G80 is a ---- CUDA core device, the NVIDIA G200 is a ---- CUDA core device, and the NVIDIA Fermi is a ---- CUDA core device.

A. 128, 256, 512

B. 32, 64, 128

C. 64, 128, 256

D. 256, 512, 1024

Answer: A

41. NVIDIA 8-series GPUs offer -------- .

A. 50-200 gflops

B. 200-400 gflops

C. 400-800 gflops

D. 800-1000 gflops

Answer: A

42. IADD, IMUL24, IMAD24, IMIN, IMAX are ---------- supported by Scalar Processors of NVIDIA GPU.

A. 32-bit ieee floating point instructions

B. 32-bit integer instructions

C. both

D. none of the above

Answer: B

43. CUDA Hardware programming model supports:
a) fully generally data-parallel archtecture;
b) General thread launch;
c) Global load-store;
d) Parallel data cache;
e) Scalar architecture;
f) Integers, bit operation

A. a,c,d,f

B. b,c,d,e

C. a,d,e,f

D. a,b,c,d,e,f

Answer: D

44. In CUDA memory model there are following memory types available:
a) Registers;
b) Local Memory;
c) Shared Memory;
d) Global Memory;
e) Constant Memory;
f) Texture Memory.

A. a, b, d, f

B. a, c, d, e, f

C. a, b, c, d, e, f

D. b, c, e, f

Answer: C

---

45. What is the equivalent of general C program with CUDA C: int main(void) { printf("Hello, World!\n"); return 0; }

A. int main (void) { kernel <<<1,1>>>(); printf("hello, world!\\n"); return 0; }

B. __global__ void kernel(void) { } int main (void) { kernel <<<1,1>>>(); printf("hello, world!\\n"); return 0; }

C. __global__ void kernel(void) { kernel <<<1,1>>>(); printf("hello, world!\\n"); return 0; }

D. __global__ int main (void) { kernel <<<1,1>>>(); printf("hello, world!\\n"); return 0; }

Answer: B

---

46. Which function runs on Device (i.e. GPU): a) __global__ void kernel (void) { } b) int main (void) { ... return 0; }

A. a

B. b

C. both a,b

D. ---

Answer: A

---

47. A simple kernel for adding two integers: __global__ void add(int *a, int *b, int *c) { *c = *a + *b; } where __global__ is a CUDA C keyword which indicates that:

A. add() will execute on device, add() will be called from host

B. add() will execute on host, add() will be called from device

C. add() will be called and executed on host

D. add() will be called and executed on device

Answer: A

---

48. If variable a is host variable and dev_a is a device (GPU) variable, to allocate memory to dev_a select correct statement:

A. cudamalloc(&dev_a, sizeof(int))

B. malloc(&dev_a, sizeof(int))

C. cudamalloc((void**) &dev_a, sizeof(int))

D. malloc((void**) &dev_a, sizeof(int))

Answer: C

49. If variable a is host variable and dev_a is a device (GPU) variable, to copy input from variable a to variable dev_a select correct statement:

A. memcpy(dev_a, &a, size);

B. cudamemcpy(dev_a, &a, size, cudamemcpyhosttodevice);

C. memcpy((void*) dev_a, &a, size);

D. cudamemcpy((void*) &dev_a, &a, size, cudamemcpydevicetohost);

Answer: B

50. Triple angle brackets mark in a statement inside main function, what does it indicates?

A. a call from host code to device code

B. a call from device code to host code

C. less than comparison

D. greater than comparison

Answer: A