CEG5201 AY2425 Exam syllabus

Prepare the following for your exam.

Nomenclature:

Slides - power-point lecture slides

It is important that you be thorough with all the material, as we have taught everything listed below in the class.

Chapter 1

- **Slides**: Go through all the slides as a revision process + Examples we have seen in the class (such as matrix multiplication, prefix-sum computation, for calculating the time-complexity of an algorithm, etc)
- Practice some simple (not overly complex!) problems on computing time-complexity; You should be able to estimate on the time-complexity, if a code snippet is given;
- Amdhal's law and its implications using fundamental definitions and computing speedups for a given problem;
- All slides on Computer clusters/ Cloud/Data center(see below the specifics for this part)

CONCEPTS & POINTS TO PONDER: Definitions of fundamental metrics like MIPS, throughput, etc; different class of architectures; Understanding of Issues & Implementation options for Embedded Systems, Must be able to compare and argue on different hardware platforms w.r.t performance related issues, Flynn's taxonomy – identifying architectures for different class of problems; time-space complexity – definitions and problems (matrix vector multiplication on parallel machines; prefix-sum computation, ***etc***); Given a code segment, you must be able to compute the time-complexity; Basic Amdahl's law and its interpretation; On different considerations while designing a compute cluster, different types of resource sharing in clusters, basic understanding of a cloud platform- issues and challenges, hierarchical network model ~~and computation of probability of paths to the cloud layer in the network~~. No need to prepare on Annexure material.

Chapter 2

- **Slides**: Parallel programming concepts – flow dependencies, Bernstein't criteria, s/w-h/w mismatch, KL process on effect of granularity, Slides on the topic - Effect of granularity on the time performance–time non-overlapped model ~~and time overlapped model~~; Multistage Interconnection Networks - all types of MINS discussed in the class - CLOS, Shuffle-Exchange, Baseline network(destination tag routing); Non-blocking criteria Lemma and its proof for CLOS; Parallel programming models discussed in the class - Different types of multi-core processors, SMTs vs Multi-core, problem solving using single threaded, MPI way, and SM way – You need to understand the basic difference between MPI & SM way of writing codes and decide the function of barriers and where to use them, given a code snippet;

CONCEPTS & POINTS TO PONDER: Identifying all flow dependencies in a code; Bernstein's conditions- given a specific hardware and a program, we must be able to compute and identify the mismatch; KL procedure on a given program graph; [code segment -> Bernstein's conditions -> Mapping on a hardware using KL approach with a given granularity]; MINS – all MINS we have dealt in the class – Non-blocking criteria Lemma and its proof for CLOS; Processing Time versus Granularity – time non-overlapped models ~~+ time overlapped models~~ - their performance w.r.t R/C ratio; Different types of multi-core processors, SMTs vs Multi-core, problem solving using single threaded, MPI way, and SM way, understanding the role of barrier directives. No need to prepare on Annexure material, if present.

Chapter 3 Modern Processor Architectures

- **Slides**: All the slides are important;

CONCEPTS & POINTS TO PONDER: Basic linear pipeline working mechanism, CISC/RISC processor systems, Superscalar, Super-pipelining and hybrid processors; Given a set of instructions you should be able to identify pipeline hazards (RAW/WAR/WAW); In-order / OOO pipeline execution computations and hazards; Use of register windows in a RISC processor; You must be able to write a pseudo-code for a given VLIW processor following an example dealt in the class and compute the speed-up, refer to the annexure on VLIW; All 3 types of multi-threaded processors and their style of working, computing the minimum number of threads in each category (as discussed in the class); No need to prepare on Annexure materials using DP formulations and solutions.

Chapter 4 Pipeline Architectures

- **Slides & concepts/points to ponder –** Linear pipeline, Non-Linear Pipeline, determining the state-space for a given non-linear pipeline, deriving the MAL for a given CV and/or a NL pipeline, Speed-up loss calculations for single cache memory (for data and instruction) systems; Speed-up for Superscalar, Super-pipeline, and superscalar-super pipeline derivations); Identifying hazards, In-order and out of order execution solutions, operand forwarding techniques, "load use" conditions, etc., for a given code snippet following all the examples we have done in the lectures, computing CPI for specific real-life pipelined processors – Refer to the annexure material we have discussed;

Chapter 5 GPU Programming

- **Slides -** It is sufficient if you use the slides discussed in the class for this chapter; You must be able to compute the time complexity, given a description of the problem, number of threads used, number of cores used, pattern of computation, etc. You must understand the sequence of steps that a CPU takes to execute on a GPU while solving a problem; Understand the vector addition example done in the lecture; On software architecture, you need to understand what constitutes a Block, a SM, and a Warp, etc., must be understood, as dealt in the class; Discussion on matrix multiplication on GPU on total time of processing;

Chapter 6 Memory Systems

- **Slides**: Follow the description below

CONCEPTS & POINTS TO PONDER: Memory characteristics, Basic methods of dealing with Cache/MM read/write methods; Basic problem solving using Direct/associative/set-associative caches; LRU and its variants (if defined any); LFU, FIFO & Belady's anomaly; Working sets definition – Problems – given a trace/application, you must be able to identify the LRU, LFU, and FIFO blocks and show the memory map when using any of the above 3 mapping techniques; Compute the number of faults for FIFO, LRU, LFU, Optimal algorithms; Shared memory organizations – higher/lower order interleaving techniques – concept and their use; Use the lecture slides for Cache coherence – methods to resolve using Snoopy bus protocols ~~and for directory-based protocols; Example ("load from mem to reg") demonstrated in the lecture would suffice; Virtual memory technology slides;~~

Note: As Section A of the Question paper may involve objective/multiple-choice type of questions/solving small problems, it is recommended that you visit all the above contents carefully.

Exam Paper Format

Time: 90 Minutes [ONE A4 size **handwritten** cheat sheet on both the sides is allowed]

Section A – 10 x 3 = 30 Marks
Time estimate to complete this section: ~ 3-5 mins per part ~= 30-40-50 mins

Section B – 2 x 10 = 20 Marks
Time estimate to complete this section: ~ 20 mins per quest ~= 40 mins

Important remark: A **penalty of 5 marks** will be imposed for unreadable, unclear handwriting and presentation on the whole;

Total Final (written exam) = 50 Marks; CA = 50 Marks

WRITING YORU ANSWERS:

- Answers MUST BE WRITTEN in the space provided under each question in the question paper and NO answer booklets are provided;
- Three (3) BLANK SHEETS will be given to you for your workings and use that space (6 sides) to do any calculations but you must capture clear answers in the space provided in the question paper. We will collect back the workings separately. No extra sheets will be provided;

===================

On Cheat sheet (1 A4 size – both sides can be used)

- It has to be HANDWRITTEN
- You cannot write on any device and take a print-out

NOTE - We will not be allowing cheat sheets that violate the above criteria.

Important dates to track:

- Nov 15 – Practice problems solutions for selected exercises will be released
- Nov 22 – Self-test will be released! (with solutions for Sect A problems) - Take this timed test and evaluate yourself on Section A type questions.

Note - Before you take the test, prepare your cheat-sheet and see if it is sufficient and amend any other info after your test.

Section A – Similar to your Self-test with reasoning and may have small problems solving;

Section B – Typically like the Practice Problems you had solved in almost every chapter!