Computer Science Department
cs.salemstate.edu

CSC 445 Parallel Processing **4 cr.**

Instructor: TBA **Office:** location **Phone:** (978) 542-extension
email: TBA@salemstate.edu **Office Hours:** days and times

Section	Time	Room	Final Exam
nn	days and times	location	date and time
Lnn	days and times	location	

Catalog description:

This course deals with the hardware and software aspects of multiprocessor systems (two or more processors in use simultaneously). The reasons for using such systems, including processor speed limitations and non-uniform capabilities of processors or computers, are examined. Basic hardware concepts such as machine models, parallel programs, networks, and performance and scalability are discussed. Relevant compiler and operating system concepts, programming models, and program development methodologies are introduced. Three lecture hours and one scheduled laboratory hour per week, plus additional laboratory work outside of class.

Prerequisites: CSC 115 and CSC 295.

Goals:

The goals for this course include the following:

CG01: to provide an introduction to the conceptual basis of parallel processing that is independent of any particular technology;
CG02: to introduce several modern parallel programming environments;
CG03: to illustrate the solution of typical numerical and graphical problems in one or more of these environments.

Objectives:

Upon successful completion of this course the student will have:

CO01: properly defined parallel processing and given examples of problems for which parallelization is appropriate;
CO02: presented a discussion of several past or present parallel machines;
CO03: programmed a heat flow problem or other applicable problem on a sequential machine and subsequently implemented that problem as a parallel program;
CO04: programmed a clustered parallel machine made up of PCs networked together running PVM or another parallel package.

Student Outcome vs. Course Objectives matrix

Student Outcome (condensed form)	CO01	CO02	CO03	CO04
SO-1	✓	✓	✓	✓
SO-2	✓		✓	✓
SO-3			✓	✓
SO-4				
SO-5			✓	✓
SO-6	✓	✓	✓	✓

Note:

SO-1 Analyze a complex computing problem and to apply principles of computing and other relevant disciplines to identify solutions.

SO-2 Design, implement, and evaluate a computing-based solution to meet a given set of computing requirements in the context of the program's discipline.
SO-3 Communicate effectively in a variety of professional contexts.
SO-4 Recognize professional responsibilities and make informed judgements in computing practice based on legal and ethical principles.
SO-5 Function effectively as a member or leader of a team engaged in activities appropriate to the program's discipline.
SO-6 Apply computer science theory and software development fundamentals to produce computing-based solutions.

Topics:

- parallel computer models
  - shared memory
  - distributed memory
  - SIMD and MIMD
  - PRAM, VLSI, and LogP models
- program and network properties **NC2(2)**
  - dependencies
  - partitioning
  - control flow vs. data flow
  - interconnect methods
- scalable performance
- parallel programming models AL11(10) (not core)
  - shared variable model
  - message passing model
  - data flow model
- parallel program design **AL4(5), SE4(2)**
  - partitioning
  - granularity
  - mapping techniques (indexing, hashing)
  - load balancing techniques (bin packing, randomization, pressure models)
- parallel languages
  - Fortran 90
  - Parallel C/C++
  - Ada95
- program development environments GV11(1), CN4(3) (not core)
- parallel operating systems

Case studies and laboratory exercises:

There will be 4 to 6 design and programming assignments in which students will be asked to implement selected techniques in parallel programming. Assignments will be work-alone or group projects depending on the nature of the assignment and equipment availability. There will also be periodic written homework assignments.

Problems will be assigned for solution in some of the following environments:

- PVM (parallel virtual machine) on a PC LAN
- concurrent programming on a single-processor machine
- shared-memory parallel processing on a multiprocessor UNIX system
- use of (part of) a massively parallel processor (MPP) via Internet access

The examples studied in class and used for programming projects will be drawn from the following list or from similar projects:

- multiplication of large matrices
- parallel solution to the Producer/Consumer Problem
- parallel sorting and searching
- numerical analysis techniques such as Simpson=s Rule for numeric integration
- numerical solution to the wave equation

Careful attention will be paid to design and implementation details.

Grading:

The course grade will be determined using the following approximate weights: laboratory reports - 40% total; written homework - 10%; two one-hour exams - 10% each; final examination - 20%.

Course Objective / Assessment Mechanism matrix

	Homework	Tests	Presentation	Project	Final Examination
CO1	✓	✓		✓	✓
CO2		✓	✓		✓
CO3			✓	✓	
CO4				✓	

Bibliography:

Topic references

Andrews, Greg. **Concurrent Programming: Principles and Practice**. Addison-Wesley, 1991.

Chandy, K. Mani & Taylor, Stephen. **An Introduction to Parallel Programming**. Jones and Bartlett, 1992.

Foster, Ian. **Designing and Building Parallel Programs: Concepts and Tools for Parallel Software Engineering**. Addison-Wesley, 1995.

Herlihy, Maurice and Nir Shavit. **The Art of Multiprocessor Programming.** Morgan Kaufman 2008

Hwang, Kai. **Advanced Computer Architecture: Parallelism, Scalability, Programmability**. McGraw-Hill, 1993.

Jordan, Harry F., and Gita Alaghband. **Fundamentals of Parallel Processing.** Prentice-hall, 2002

Kumar, Vipin; Grama, Ananth; Gupta, Anshul; Karypis, George. **An Introduction to Parallel Computing: Design and Analysis of Algorithms**. **Second Edition.** Addison Wesley, 2003.

Nevison, C. H., et. al. **Laboratories for Parallel Computing**. Jones and Bartlett, 1994.

Pacheco, Peter S. **An Introduction to Parallel Programming.** Morgan Kaufman, 2011

Sandhu, Harprit. **Programming the Propeller with Spin: A Beginners Guide to Parallel Processing.** Tab Electronics, 2010

Parallax Corporation. **The Propeller Chip: Operation and Programming.** 2008.

Language references

Parallax Corporation. **The SPIN Reference Manual.** 2008.

Academic Integrity Statement:

"Salem State University assumes that all students come to the University with serious educational intent and expects them to be mature, responsible individuals who will exhibit high standards of honesty and personal conduct in their academic life. All forms of academic dishonesty are considered to be serious offences against the University community. The University will apply sanctions when student conduct interferes with the University primary responsibility of ensuring its educational objectives." Consult the University catalog for further details on Academic Integrity Regulations and, in particular, the University definition of academic dishonesty.

The Academic Integrity Policy and Regulations can be found in the University Catalog and on the University website (http://catalog.salemstate.edu/content.php?catoid=13&navoid=1295#Academic_Integrity). The formal regulations are extensive and detailed - familiarize yourself with them if you have not previously done so. A concise summary of and direct quote from the regulations: "Materials (written or otherwise) submitted to fulfill academic requirements must represent a student's own efforts". *Submission of other's work as one's own without proper attribution is in direct violation of the University's Policy* and will be dealt with according to the University's formal Procedures. *Copying without attribution is considered cheating in an academic environment - simply put,* ***do not do it!***

University-Declared Critical Emergency Statement:

In the event of a university-declared emergency, Salem State University reserves the right to alter this course plan. Students should refer to www.salemstate.edu for further information and updates. The course attendance policy stays in effect until there is a university-declared critical emergency.

In the event of an emergency, please refer to the alternative educational plans for this course, which will be distributed via

standing class communication protocols. Students should review the plans and act accordingly. Any required material that may be necessary will have been previously distributed to students electronically or will be made available as needed via email and/or Internet access.

Equal Access Statement:

"Salem State University is committed to providing equal access to the educational experience for all students in compliance with Section 504 of The Rehabilitation Act and The Americans with Disabilities Act and to providing all reasonable academic accommodations, aids and adjustments. **Any student who has a documented disability requiring an accommodation, aid or adjustment should speak with the instructor immediately.** Students with Disabilities who have not previously done so should provide documentation to and schedule an appointment with the Office for Students with Disabilities and obtain appropriate services."

Note: This syllabus represents the intended structure of the course for the semester. If changes are necessary, students will be notified in writing and via email.