UNIVERSITY OF GUAM
COURSE OUTLINE FORM

College: College of Natural and Applied Sciences Course Number: CS*200

Course Title: Computer Applications and Lab Credit Hours: 3 crs.

Date of Final Approval: Semester Offered: F/SP

Course counts as X general education requirement
X part of Computer Science/CIS major program
____elective
____NDU

1. CATALOG DESCRIPTION

This course covers preparation, storage, and processing of data, documents, and illustrations; graphing, manipulating and simple analysis of data; computer-to-computer communications and file transfers; use of UOG network resources; and introduction to computer languages. Prerequisite: MA*085 Level II or Math Placement or Consent of the Instructor.

2. COURSE CONTENT

This course covers the following topics: microcomputer operations, graphic user operating system; hands-on application packages – word processing, spreadsheets, database management systems, business graphics, data communications, and computer languages.

3. RATIONALE FOR THE COURSE

This is a required course for the proposed Computer Science Degree Program and Computer Information System Program. It is designed to introduce and prepare students in computer applications and lab for the very rigorous upper division computer science courses.

4. SKILLS AND BACKGROUND REQUIRED OR EXPECTED

None.

5. TEACHING METHODOLOGIES AND ANTICIPATED CLASS SIZE

Standard, 20 students.

6. LEARNING OBJECTIVES FOR STUDENTS

- Describe the major components and properties of all modern computer systems.
- Using MS Word, create a well-formatted research paper with outline, embedded table, graphic illustration, and references.
- Using MS Excel, create a well-formatted spreadsheet to calculate a cash-flow student budget.
- Using MS PowerPoint, create a well-formatted presentation corresponding to the outline for the above research paper (in SLO #2).
- Using HTML, create well-formatted, linked WebPages.

Note: With Program Faculty Consultation, an instructor may add additional SLOs to the above Program Faculty approved SLOs.

7. METHODS OF EVALUATION

Standard: Exercises, projects, quizzes, and exams.

8. METHODS FOR STUDENT LEARNING OUTCOMES ASSESSMENT:

Depending on Instructor and Program Faculty, any one or more of the following may be selected: Pre/Post Test, Course embedded questions; Standardized exams; Portfolio Evaluation; Direct Observation; and Capstone Course Evaluation.

9. REQUIRED AND RECOMMENDED TEXTS AND/OR STUDY GUIDES:

Teachers Discovering Computers: Integrating Technology and Digital Media in the Classroom, 5th edition, by Shelly, Cashman, et. al. Course Technology.
Lab Book: Microsoft Office 2003, 2nd edition, by Shelly, Cashman, Vermaat. Course Technology.

10. SUBSEQUENT COURSES:

CS*201.

11. ADDITIONAL COURSE DESCRIPTORS, IF ANY:

None.

The Calendar of Assignments, Assessment Project, a Statement Concerning the "Americans with Disabilities Act" (ADA) Accommodations for Students, Attendance and Grading Policies are to be included in the course syllabus.

UNIVERSITY OF GUAM
COURSE OUTLINE FORM

College: College of Natural and Applied Sciences Course Number: CS*201

Course Title: Programming I Credit Hours: 4 crs.

Date of Final Approval: Semester Offered: F/SP

Course counts as _x_ general education requirement
_x_ part of Computer Science/CIS major program
____ elective
____ NDU

2. CATALOG DESCRIPTION

This course introduces high level computer programming languages with emphasis on program design, coding, debugging, testing, and proper documentation, with applications. Prerequisites: CS*200, MA*161a or higher.

2. COURSE CONTENT

This course introduces a number of topics fundamental to programming in high-level computer languages. These topics include basic concepts about hardware and software in general; basic input/output operations; basic data types—integers and real numbers; basic arithmetic operations and expressions; assignment statements; ways to control program logic flow by using conditional expressions and statements and looping structures for repetition. Also covered are other data types like characters, strings, and arrays; and subprograms: functions, procedures, and subroutines.

3. RATIONALE FOR THE COURSE

This will give students a sound foundation to understand and master the basic, archetypal forms underlying the construction of programs in all major high-level programming languages. With this foundation, students should be able to transition more easily into the more powerful, albeit more terse, forms common to programming and so important now to our world of computers.

4. SKILLS AND BACKGROUND REQUIRED OR EXPECTED

Since computer program requirements are often described by using mathematical relationships, the student should have a sound mathematical understanding, at least through the level of MA*161/MA*110, Fundamentals of College Mathematics/Finite Mathematics; and a thorough understanding of basic computer operations as from CS*200 Computer Applications with Lab.

5. TEACHING METHODOLOGIES AND ANTICIPATED CLASS SIZE

The course composed of 16 weeks of 4 50-minute lectures per week; 20 students are expected.

6. LEARNING OBJECTIVES FOR STUDENTS

- Construct appropriate I/O statements.
- Construct assignment statements.
- Construct conditional statements.
- Construct looping statements.

- Design, code, test and debug programs, utilizing these constructs to create solutions for simple problems.

Note: With Program Faculty Consultation, an instructor may add additional SLOs to the above Program Faculty approved SLOs.

7. METHODS OF EVALUATION

Standard: Exercises, projects, quizzes, and exams.

8. METHOD FOR STUDENT LEARNING OUTCOMES ASSESSMENT:

Depending on Instructor and Program Faculty, any one or more of the following may be selected: Pre/Post Test, Course embedded questions; Standardized exams; Portfolio Evaluation; Direct Observation; and Capstone Course Evaluation.

9. REQUIRED AND RECOMMENDED TEXTS ABD/OR STUDY GUIDES:

Internet and the WWW How to Program, 3rd edition, by Deitel, H. ISBN 0131450913, Pearson Education, 2003.

10. SUBSEQUENT COURSES:

CS*202.

11. ADDITIONAL COURSE DESCRIPTORS, IF ANY:

None.

The Calendar of Assignments, Assessment Project, a Statement Concerning the "Americans with Disabilities Act" (ADA) Accommodations for Students, Attendance and Grading Policies are to be included in the course syllabus.

UNIVERSITY OF GUAM
COURSE OUTLINE FORM

College: College of Natural and Applied Sciences Course Number: CS*202

Course Title: Programming II Credit Hours: 4 crs.

Date of Final Approval: Semester Offered: F/SP

Course counts as __x__ general education requirement
__x__ part of Computer Science/CIS major program
____elective
____NDU

3. CATALOG DESCRIPTION

This course introduces computer concepts and problem-solving methods. Topics include algorithms, data types, objects, classes, encapsulation, and exception handling. Emphasis is placed on structured program design. Prerequisite: CS*200, CS*201, MA*161a or higher.

2. COURSE CONTENT

This course covers the following topics: introduction to problem solving, algorithm development, procedural and data abstraction; program design, coding, debugging, testing and documentation. CS*202 teaches computer programming an object-oriented language.

3. RATIONALE FOR THE COURSE

This course is required for the Computer Science Degree Program.

4. SKILLS AND BACKGROUND REQUIRED OR EXPECTED

CS*200, CS*201.

5. TEACHING METHODOLOGIES AND ANTICIPATED CLASS SIZE

Standard, 20 students.

6. LEARNING OBJECTIVES FOR STUDENTS

- Describe the steps involved in the problem solving process.
- Utilize various design tools to develop correct solutions to problems.
- Succinctly document both the problem and the solution in given programming tasks.
- Apply object-oriented programming techniques to develop real world applications (define objects and their properties, constructing appropriate methods for each).
- Code, test and debug efficient, event-driven programs to implement solutions of simple problems in a visual development environment using the Java programming language.

Note: With Program Faculty Consultation, an instructor may add additional SLOs to the above Program Faculty approved SLOs.

7. METHODS OF EVALUATION

Standard: Exercises, Projects, Quizzes and Exams.

8. METHOD FOR STUDENT LEARNING OUTCOMES ASSESSMENT:

Standard: Exercises, Projects, Quizzes and Exams. Depending on Instructor and Program Faculty, any one or more of the following may be selected: Pre/Post Test, Course embedded questions; Standardized exams; Portfolio Evaluation; Direct Observation; and Capstone Course Evaluation.

9. REQUIRED AND RECOMMENDED TEXTS AND/OR STUDY GUIDES

Introduction to Java Programming, 6th edition, by Y. Daniel Lian, Pearson Education/Prentice Hall, 2006.

10. SUBSEQUENT COURSES

CS*305, CS*370.

11. ADDITIONAL COURSE DESCRIPTORS, IF ANY

None.

The Calendar of Assignments, Assessment Project, a Statement Concerning the "Americans with Disabilities Act" (ADA) Accommodations for Students, Attendance and Grading Policies are to be included in the course syllabus.

UNIVERSITY OF GUAM
COURSE OUTLINE FORM

College: College of Natural and Applied Sciences Course Number: CS*305

Course Title: Assembly Language and Computer Organization Credit Hours: 3 crs.

Date of Final Approval: Semester Offered: F

Course counts as x general education requirement
x part of Computer Science/CIS major program
____ elective
____ NDU

4. CATALOG DESCRIPTION

This course covers the organization and structure of the major components of modern computer; combinational circuits, sequential circuits, simulation of circuits, coding, computer organization and architecture, including a detailed study of a micro-computer or minicomputer and the use of Assembly Language. Prerequisite: CS*201 and CS*202 or Consent of Instructor.

2. COURSE CONTENT

Programming in x86 Assembly Language.

3. RATIONALE FOR THE COURSE

Required for degree.

4. SKILLS AND BACKGROUND REQUIRED OR EXPECTED

CS*200, CS*201, CS*202.

5. TEACHING METHODOLOGIES AND ANTICIPATED CLASS SIZE

Standard, 20 students.

6. LEARNING OBJECTIVES FOR STUDENTS

After completing this course, students will be able to

- Identify the function of each major component of microcomputer systems, including bits, gates, circuits, and the levels of languages used to control them (including machine, assembly, and high-level)
- Convert the various number systems (binary, octal, decimal and hexadecimal) used in computers.
- Design, code, and test programs using the basic grammar, syntax and standards of the Intel-based assembly language (16-bit and 32-bit modes) to solve real world problems.

Note: With Program Faculty Consultation, an instructor may add additional SLOs to the above Program Faculty approved SLOs.

7. METHODS OF EVALUATION

Standard: Exercises, projects, quizzes, and exams.

8. METHOD FOR STUDENT LEARNING OUTCOMES ASSESSMENT:

Depending on Instructor and Program Faculty, any one or more of the following may be selected: Pre/Post Test, Course embedded questions; Standardized exams; Portfolio Evaluation; Direct Observation; and Capstone Course Evaluation.

9. REQUIRED AND RECOMMENDED TEXTS AND/OR STUDY GUIDES

Assembly Language for Intel-Based Computers, 5th edition, by Kip R. Irvine. Prentice Hall/Pearson Education, 2006.

10. SUBSEQUENT COURSES

See catalog.

11. ADDITIONAL COURSE DESCRIPTORS, IF ANY:

None.

The Calendar of Assignments, Assessment Project, a Statement Concerning the "Americans with Disabilities Act" (ADA) Accommodations for Students, Attendance and Grading Policies are to be included in the course syllabus.

UNIVERSITY OF GUAM
COURSE OUTLINE FORM

College: College of Natural and Applied Sciences Course Number: CS*315

Course Title: Intro to File Processing and Database Management Credit Hours: 3 crs.

Date of Final Approval: Semester Offered: F/SP

Course counts as x general education requirement
x part of Computer Science/CIS major program
____ elective
____ NDU

5. CATALOG DESCRIPTION

This course covers the following concepts: Definition of file components, access methods and file operations; algorithms for efficient implementation of data structures; characteristics of bulk storage media or mainframe and microcomputer or minicomputer; introduction to database management systems. Prerequisite: CS*200, CS*201 or Consent of Instructor.

2. COURSE CONTENT

File systems and Databases
The Relational Database Model
An Introduction to SQL
Entity-Relationship Modeling
Normalization of Database Tables
Database Design
Transaction Management and Concurrency Control
Distributed Database Management Systems.

3. RATIONALE FOR THE COURSE

This is an advanced course in the theory and design of database management systems, and is intended for future computer scientists, programmers, and systems administrators, and DBA's.

4. SKILLS AND BACKGROUND REQUIRED OR EXPECTED

CS*200, CS*201.

5. TEACHING METHODOLOGIES AND ANTICIPATED CLASS SIZE

Standard, 20 students.

6. LEARNING OBJECTIVES FOR STUDENTS

- Design, develop and implement small professional databases using a specified relational database management system (DBMS).
- Correctly demonstrate the steps for data normalization when designing a database.
- Utilize the SQL (Structured Query Language) fluently for application development.
- Describe and distinguish the features of Object-Oriented DBMS and Distributed DBMS.
- Describe and demonstrate database administration, security, transaction failure and recovery.

Note: With Program Faculty Consultation, an instructor may add additional SLOs to the above Program Faculty approved SLOs.

7. METHODS OF EVALUATION

Standard: Exercises, projects, quizzes, and exams.

8. METHOD FOR STUDENT LEARNING OUTCOMES ASSESSMENT:

Depending on Instructor and Program Faculty, any one or more of the following may be selected: Pre/Post Test, Course embedded questions; Standardized exams; Portfolio Evaluation; Direct Observation; and Capstone Course Evaluation.

9. REQUIRED AND RECOMMENDED TEXTS AND/OR STUDY GUIDES

Database Management Systems, 3rd revised edition, by Gerald V. Post, McGraw-Hill, 2004.
Lab Book: Access 2003: Comprehensive Concepts and Techniques, Coursecard, by Shelly, Cashman and Pratt, Cengage Learning.

10. SUBSEQUENT COURSES

CS*360.

11. ADDITIONAL COURSE DESCRIPTORS, IF ANY:

None.

The Calendar of Assignments, Assessment Project, a Statement Concerning the "Americans with Disabilities Act" (ADA) Accommodations for Students, Attendance and Grading Policies are to be included in the course syllabus.

UNIVERSITY OF GUAM
COURSE OUTLINE FORM

College: College of Natural and Applied Sciences Course Number: CS*360

Course Title: Introduction to Operating Systems Credit Hours: 4 crs.

Date of Final Approval: Semester Offered: F/SP

Course counts as _x_ general education requirement
_x_ part of Computer Science/CIS major program
____ elective
____ NDU

6. CATALOG DESCRIPTION

This course covers the following concepts: Operating system history and services; file systems; memory management; process management-concurrent processes, communication, semaphores, monitors, deadlocks, resource management processor and disk scheduling; security and protection systems. It meets for three hours lecture and three hours laboratory weekly. Prerequisite: CS*202.

2. COURSE CONTENT

This course covers: process synchronization; language mechanisms for concurrency; deadlock; virtual memory; distributed systems; distributed concurrency; control, deadlock, and recovery; computer security; and quenching models of computer systems.

3. RATIONALE FOR THE COURSE

This course is required for the Computer Science Degree Program.

4. SKILLS AND BACKGROUND REQUIRED OR EXPECTED

CS*200, CS*201, CS*202.

5. TEACHING METHODOLOGIES AND ANTICIPATED CLASS SIZE

Standard, 20 students.

6. LEARNING OBJECTIVES FOR STUDENTS

- Describe the history and importance of the operating system (OS) in a computer system.
- Completely describe the five major tasks of an OS (i.e. process management, memory management, file management, I/O device management, and network management.
- Describe the roles and importance of computer system management and security.
- Describe the features, strengths, and weaknesses of different operating systems, such as batch, interactive, real-time, embedded, and parallel systems.
- Install, configure, and manage the MS-Windows, UNIX and Linux Operating Systems. Simulate and optimize some component parts of each OS.

Note: With Program Faculty Consultation, an instructor may add additional SLOs to the above Program Faculty approved SLOs.

7. METHODS OF EVALUATION

Standard: Exercises, projects, quizzes, and exams.

8. METHOD FOR STUDENT LEARNING OUTCOMES ASSESSMENT:

Depending on Instructor and Program Faculty, any one or more of the following may be selected: Pre/Post Test, Course embedded questions; Standardized exams; Portfolio Evaluation; Direct Observation; and Capstone Course Evaluation.

9. REQUIRED AND RECOMMENDED TEXTS AND/OR STUDY GUIDES

Understanding Operating Systems, 4th edition, by Flynn & McHoes, Cengage Learning, 2005.
Lab Book: Microsoft Windows XP: Comprehensive Concepts and Techniques by Shelly, Cashman and Forsythe, Service Pack 2, Cengage Learning, 2005.

10. SUBSEQUENT COURSES

CS*403.

11. ADDITIONAL COURSE DESCRIPTORS, IF ANY:

None.

The Calendar of Assignments, Assessment Project, a Statement Concerning the "Americans with Disabilities Act" (ADA) Accommodations for Students, Attendance and Grading Policies are to be included in the course syllabus.

UNIVERSITY OF GUAM
COURSE OUTLINE FORM

College: College of Natural and Applied Sciences Course Number: CS*365

Course Title: Computer Architecture Credit Hours: 3 crs.

Date of Final Approval: Semester Offered: SP/Odd Years

Course counts as _x_ general education requirement
_x_ part of Computer Science/CIS major program
____ elective
____ NDU

7. CATALOG DESCRIPTION

This course is an in-depth analysis of computer hardware architecture. Other topics may include parallel or multi-core hardware architectures and performance analysis. Prerequisite: CS*202 and CS*305.

2. COURSE CONTENT

This course covers: the scope of computer architecture; technologic framework; the design process; uniprocessors; register machines architecture and microprogramming: the exploitation of stacks; language-directed architectures; the "RISC" style; and aspects of memory.

3. RATIONALE FOR THE COURSE

This course is required for the Computer Science Degree Program.

4. SKILLS AND BACKGROUND REQUIRED OR EXPECTED

CS*202 and CS*305.

5. TEACHING METHODOLOGIES AND ANTICIPATED CLASS SIZE

Standard, 20 students.

6. LEARNING OBJECTIVES FOR STUDENTS

- Identify and describe all the major components of computer systems and CPU's.
- Calculate and compare the duration of basic operations with machine cycles.
- Describe the basic steps of a complete machine cycle and the common ways to speed up processor execution.
- Describe how to construct CPU components from logic gates utilizing the basic concepts of digital electronics.
- Compare and contrast RISC vs. CISC, and single vs. multi-core CPU Architectures.

Note: With Program Faculty Consultation, an instructor may add additional SLOs to the above Program Faculty approved SLOs.

7. METHODS OF EVALUATION

Standard: Exercises, projects, quizzes, and exams.

8. METHOD FOR STUDENT LEARNING OUTCOMES ASSESSMENT:

Depending on Instructor and Program Faculty, any one or more of the following may be selected: Pre/Post Test, Course embedded questions; Standardized exams; Portfolio Evaluation; Direct Observation; and Capstone Course Evaluation.

9. REQUIRED AND RECOMMENDED TEXTS AND/OR STUDY GUIDES

Principles of Computer Architecture by Murdocca and Heuring, ISBN 0201436647. Pearson Education/Prentice Hall, 2000.

10. SUBSEQUENT COURSES

CS*380.

11. ADDITIONAL COURSE DESCRIPTORS, IF ANY:

None.

The Calendar of Assignments, Assessment Project, a Statement Concerning the "Americans with Disabilities Act" (ADA) Accommodations for Students, Attendance and Grading Policies are to be included in the course syllabus.

UNIVERSITY OF GUAM
COURSE OUTLINE FORM

College: College of Natural and Applied Sciences Course Number: CS*370

Course Title: Data Structures and Algorithm Analysis Credit Hours: 3 crs.

Date of Final Approval: Semester Offered: SP

Course counts as _x_ general education requirement
_x_ part of Computer Science/CIS major program
____ elective
____ NDU

8. CATALOG DESCRIPTION

The course covers the following concepts: Basic data structures (linked lists, stacks, and trees); graphs, search path and spanning trees; searching; memory management, hashing, dynamic storage allocation; integration of data structures into system design. Prerequisite: CS*202.

2. COURSE CONTENT

Objects and classes, templates, inheritance, algorithm analysis, sorting algorithms, randomization algorithms, stacks, graphs and paths, queues, linked lists, trees, binary search trees, hash tables.

3. RATIONALE FOR THE COURSE

This is the definitive programming course, codifying the various techniques utilized in successful code generation.

4. SKILLS AND BACKGROUND REQUIRED OR EXPECTED

CS*201 and CS*202.

5. TEACHING METHODOLOGIES AND ANTICIPATED CLASS SIZE

Standard, 20 students.

6. LEARNING OBJECTIVES FOR STUDENTS

- Describe the operation, implementation and performance of fundamental algorithms and data structures, and the relative merits and suitability of each for various applications.
- Design, implement, and test efficient software solutions for various application areas using appropriately selected algorithms and data structures (including especially linked-lists, stacks, and trees).
- Analyze data structures and algorithms, by comparing and evaluating them with respect to time and space requirements, in order to make the most appropriate design choices for various application areas.
- Motivate and explain efficient programming concepts, relevant alternatives and decision recommendations, in written form, to IT specialists.
- Apply relevant standards and ethical considerations to the design and implementation of efficient software solutions.

Note: With Program Faculty Consultation, an instructor may add additional SLOs to the above Program Faculty approved SLOs.

7. METHODS OF EVALUATION

Standard: Exercises, projects, quizzes, and exams.

8. METHOD FOR STUDENT LEARNING OUTCOMES ASSESSMENT:

Depending on Instructor and Program Faculty, any one or more of the following may be selected: Pre/Post Test, Course embedded questions; Standardized exams; Portfolio Evaluation; Direct Observation; and Capstone Course Evaluation.

9. REQUIRED AND RECOMMENDED TEXTS AND/OR STUDY GUIDES

Data Structures and Algorithms in Java, Latest Edition, by Peter Drake, Prentice Hall/Pearson Education

10. SUBSEQUENT COURSES

CS*360.

11. ADDITIONAL COURSE DESCRIPTORS, IF ANY:

None.

UNIVERSITY OF GUAM
COURSE OUTLINE FORM

College: College of Natural and Applied Sciences Course Number: CS*380

Course Title: Organization of Programming Languages Credit Hours: 3 crs.

Date of Final Approval: Semester Offered: F/Odd Years

Course counts as _x_ general education requirement
_x_ part of Computer Science major program
____ elective
____ NDU

9. CATALOG DESCRIPTION

The course examines general concepts of programming languages; scope and binding rules, applications and implementation of language concepts, including the study of two or more languages. Prerequisite: CS*201 and CS*202.

2. COURSE CONTENT

The course discusses: A survey of the history and development of programming languages by comparing the common features of programming languages including the modern ideas of modularity and abstract data types.

3. RATIONALE FOR THE COURSE

This course is required for the proposed Computer Science Degree Program..

4. SKILLS AND BACKGROUND REQUIRED OR EXPECTED

CS*201, CS*202, and CS*370.

5. TEACHING METHODOLOGIES AND ANTICIPATED CLASS SIZE

Standard, 20 students.

6. LEARNING OBJECTIVES FOR STUDENTS

- Identify and describe the four major families of programming languages.
- Design, code, test and debug a program using at least one Imperative language.
- Design, code, test and debug a program using at least one Functional language.
- Design, code, test and debug a program using at least one Logic language.
- Design, code, test and debug a program using at least one Object-oriented language.

Note: With Program Faculty Consultation, an instructor may add additional SLOs to the above Program Faculty approved SLOs.

7. METHODS OF EVALUATION

Standard: Exercises, projects, quizzes, and exams.

8. METHOD FOR STUDENT LEARNING OUTCOMES ASSESSMENT:

Depending on Instructor and Program Faculty, any one or more of the following may be selected: Pre/Post Test, Course embedded questions; Standardized exams; Portfolio Evaluation; Direct Observation; and Capstone Course Evaluation.

9. REQUIRED AND RECOMMENDED TEXTS AND/OR STUDY GUIDES

Study of Programming Languages, by Ryan Stansifer, latest edition, Prentice Hall.

10. SUBSEQUENT COURSES

CS*410.

11. ADDITIONAL COURSE DESCRIPTORS, IF ANY:

None.

The Calendar of Assignments, Assessment Project, a Statement Concerning the "Americans with Disabilities Act" (ADA) Accommodations for Students, Attendance and Grading Policies are to be included in the course syllabus.

UNIVERSITY OF GUAM
COURSE OUTLINE FORM

College: College of Natural and Applied Sciences Course Number: CS*403

Course Title: Data Communication and Networking Credit Hours: 3 crs.

Date of Final Approval: Semester Offered: SP/Odd Years

Course counts as x general education requirement
x part of Computer Science/CIS major program
____ elective
____ NDU

10. CATALOG DESCRIPTION

The course covers the following concepts: Data communication and network concepts; protocol and standards; distributed computing; local and wide area networks. Prerequisite: CS*370.

2. COURSE CONTENT

Data Communication and Network Concepts.

3. RATIONALE FOR THE COURSE

This course is required for the proposed Computer Science Degree Program.

4. SKILLS AND BACKGROUND REQUIRED OR EXPECTED

CS*202, CS*360, CS*370.

5. TEACHING METHODOLOGIES AND ANTICIPATED CLASS SIZE

Standard, 20 students are expected.

6. LEARNING OBJECTIVES FOR STUDENTS

- Describe the fundamental concepts and principles of telecommunications systems (e.g. configurations, protocols, OSI and TCP/IP models), and data and signal conversion for data communications.
- Describe the basic principles and structures of computer networks (LAN, MAN, PAN, WAN, Internet and World Wide Web).
- Describe the principles of network design, management, and security.
- Use actual hardware and software for network connections, administration, and multiplexing, along with error detection and error correction to master the skills of network operating systems (NOS, e.g. MS Windows or LINUX) through the hands-on projects. The skills attained through the manipulation of NOS will be applicable to network administration of a client-server system utilizing the NOS.

Note: With Program Faculty Consultation, an instructor may add additional SLOs to the above Program Faculty approved SLOs.

7. METHODS OF EVALUATION

Standard: Exercises, projects, quizzes and exams.

8. METHOD FOR STUDENT LEARNING OUTCOMES ASSESSMENT:

Depending on Instructor and Program Faculty, any one or more of the following may be selected: Pre/Post Test, Course embedded questions; Standardized exams; Portfolio Evaluation; Direct Observation; and Capstone Course Evaluation.

9. REQUIRED AND RECOMMENDED TEXTS AND/OR STUDY GUIDES

Data Communications and Computer Networks, 4th edition, by Curt M. White, Cengage Learning, . ISBN 1418836109, 2006.

10. SUBSEQUENT COURSES

CS*431.

11. ADDITIONAL COURSE DESCRIPTORS, IF ANY:

None.

UNIVERSITY OF GUAM
COURSE OUTLINE FORM

College: College of Natural and Applied Sciences Course Number: CS*410

Course Title: Compiler Design and Construction Credit Hours: 3 crs.

Date of Final Approval: Semester Offered: SP/Even Years

Course counts as x general education requirement
x part of Computer Science major program
____ elective
____ NDU

11. CATALOG DESCRIPTION

The course covers the following concepts: Syntax and semantics of programming languages, lexical analysis, parsing techniques, run-time storage management, symbol table organizations; introduction to semantic routine, intermediate codes, interpreters, error recovery techniques; code optimization and generation; compiler generators. Prerequisite: CS*202, CS*305, CS*370, CS*380.

2. COURSE CONTENT

Compiler Theory and Algorithms.

3. RATIONALE FOR THE COURSE

This course is required for the proposed Computer Science Degree Program.

4. SKILLS AND BACKGROUND REQUIRED OR EXPECTED

CS*202, CS*305, CS*370, CS*380.

5. TEACHING METHODOLOGIES AND ANTICIPATED CLASS SIZE

Standard, 20 students.

6. LEARNING OBJECTIVES FOR STUDENTS

- Describe the purposes of and differences between compilers, interpreters and assemblers.
- Construct Regular Expressions.
- Apply the theories of finite automata and context-free grammars to construct a Parser.
- Apply the theories of finite automata and context-free grammars to construct a Lexical Analyzer.
- Generate Object code for given Source Code in a specified language.

Note: With Program Faculty Consultation, an instructor may add additional SLOs to the above Program Faculty approved SLOs.

7. METHODS OF EVALUATION

Standard: Exercises, projects, quizzes, and exams.

8. METHOD FOR STUDENT LEARNING OUTCOMES ASSESSMENT:

Depending on Instructor and Program Faculty, any one or more of the following may be selected: Pre/Post Test, Course embedded questions; Standardized exams; Portfolio Evaluation; Direct Observation; and Capstone Course Evaluation.

9. REQUIRED AND RECOMMENDED TEXTS AND/OR STUDY GUIDES

Compiler Construction Principles and Practice, Latest Edition, K. C. Louden, Cengage Learning (ITP).

10. SUBSEQUENT COURSES

CS*431.

11. ADDITIONAL COURSE DESCRIPTORS, IF ANY:

None.

The Calendar of Assignments, Assessment Project, a Statement Concerning the “Americans with Disabilities Act” (ADA) Accommodations for Students, Attendance and Grading Policies are to be included in the course syllabus.

UNIVERSITY OF GUAM
COURSE OUTLINE FORM

College: College of Natural and Applied Sciences Course Number: CS*431

Course Title: Advanced Topics in Computing (Systems Analysis & Design) – Capstone Course Credit Hours: 3 crs.

Date of Final Approval: Semester Offered: F

Course counts as __x__ general education requirement
__x__ part of Computer Science/CIS major program
____ elective
____ NDU

12. CATALOG DESCRIPTION

This course is a study of selected topics in programming and information systems, such as computer graphics, computer modeling, informing modeling, software engineering, design and analysis of software. With different subject matter, it may be repeated for credit. Prerequisites: Junior/senior level standing in computer science/computer information systems, or consent of instructor.

2. COURSE CONTENT

Advanced, timely topics depends on instructor's choice (e.g. Systems analysis and design, computer security).

3. RATIONALE FOR THE COURSE

This course is required for the Computer Science Degree Program.

4. SKILLS AND BACKGROUND REQUIRED OR EXPECTED

As one of the final courses, very advanced skills.

5. TEACHING METHODOLOGIES AND ANTICIPATED CLASS SIZE

Standard, 20 students.

6. LEARNING OBJECTIVES FOR STUDENTS

Upon completion of this course successfully, students will be able to:

- Apply General Systems Theory to describe any system in terms of its components and sub-systems.
- Identify and describe the purpose of a specified system.
- Identify and describe the inputs, outputs, and resources of a specified system.
- Design a system to achieve its specified purpose.
- Implement the designed system to transform the inputs and resources into the desired outputs to achieve its intended specified purpose.

Note: With Program Faculty Consultation, an instructor may add additional SLOs to the above Program Faculty approved SLOs.

7. METHODS OF EVALUATION

Standard: Exercises, projects, quizzes, and exams.

8. METHOD FOR STUDENT LEARNING OUTCOMES ASSESSMENT:

Depending on Instructor and Program Faculty, any one or more of the following may be selected: Pre/Post Test, Course embedded questions; Standardized exams; Portfolio Evaluation; Direct Observation; and Capstone Course Evaluation.

9. REQUIRED AND RECOMMENDED TEXTS AND/OR STUDY GUIDES

Object-Oriented and Classical Software Engineering, 6th edition by Stephen R. Schach, McGraw-Hill. ISBN: 0072865512, 2005.

10. SUBSEQUENT COURSES

CS*492.

11. ADDITIONAL COURSE DESCRIPTORS, IF ANY:

None.

UNIVERSITY OF GUAM
COURSE OUTLINE FORM

College: College of Natural and Applied Sciences Course Number: CS*492

Course Title: Practicum in Computer Science Credit Hours: 1-3 crs.

Date of Final Approval: Semester Offered: F/SP

Course counts as _x_ general education requirement
_x_ part of Computer Science/CIS major program
____ elective
____ NDU

13. CATALOG DESCRIPTION

The course is a laboratory–oriented course involving any laboratory from microcomputers to mainframe depending on the project director administering the practicum. The mathematics computer lab, the computer center labs, or any other computer lab facility may be utilized as worksites for the course. The course provides practical experience for students majoring in computer science or management information systems. Students assist in the daily operation of the computer lab. Regular observation, counseling, and evaluation is provided by the project director. Individual and/or group conferences are required. A junior or senior level student in computer science may enroll for the course with the approval of the Division of Mathematical Sciences. Students may enroll more than once, but a maximum of 3-semester hours credit may be earned in this course.

2. COURSE CONTENT

See above Catalog description.

3. RATIONALE FOR THE COURSE

Good, practical experience.

4. SKILLS AND BACKGROUND REQUIRED OR EXPECTED

All major skills.

5. TEACHING METHODOLOGIES AND ANTICIPATED CLASS SIZE

Standard, hands-on, 20 students.

6. LEARNING OBJECTIVES FOR STUDENTS

- Trouble-shoot, repair and configure computers, servers, routers, printers, copiers, and IP telephony equipment in corporate LAN, WAN, and Wireless environments.
- Use Microsoft's Remote Install Service (RIS), and Symantec's Ghost to clone and mass duplicate corporate PC's.
- Do simple Cat5e cabling and crimping, as well as coax cabling legacy equipment.
- Use different applications for the remote control and remote updating of PCs, such as Microsoft's SMS and Remote Desktop, Dameware, and VN, as well as the corporate specific Helpdesk ticket-generated application called Vantive (used in Continental's and EDS's corporate environments).

Note: With Program Faculty Consultation, an instructor may add additional SLOs to the above Program Faculty approved SLOs.

7. METHODS OF EVALUATION

Standard: Exercises, projects, quizzes, and exams.

8. METHOD FOR STUDENT LEARNING OUTCOMES ASSESSMENT:

Depending on Instructor and Program Faculty, any one or more of the following may be selected: Pre/Post Test, Course embedded questions; Standardized exams; Portfolio Evaluation; Direct Observation; and Capstone Course Evaluation.

9. REQUIRED AND RECOMMENDED TEXTS AND/OR STUDY GUIDES

Network + Guide to Network, 4th edition, by Tamara Dean, Cengage Learning, 2005.

10. SUBSEQUENT COURSES

None.

11. ADDITIONAL COURSE DESCRIPTORS, IF ANY:

None.

The Calendar of Assignments, Assessment Project, a Statement Concerning the "Americans with Disabilities Act" (ADA) Accommodations for Students, Attendance and Grading Policies are to be included in the course syllabus.