

CIUJ: UI4111 Computer Systems

SUT: HIT1015 Computer Systems

Credit Points 12.5

Contact :Hours 36 Hours

Duration 1 Semester

Campus :CIUJ

Prerequisites Nil

Corequisites Nil

Teaching Method

Lecture (2 Hours per Week), Laboratory (1 Hour per Week)

Assessment

Assignments, Examinations

Aims & Objectives

- * To understand the fundamental concepts of computer systems.
- * To gain experience in assembly language programming.

Content

- * Functions and components of computers.
- * Data representation.
- * Computer logic.
- * Elementary assembly language programming.
- * Secondary storage and input/output devices.

Reading Materials

Tocci, R., Digital Systems, Principles and Applications, 8th edn, Prentice-Hall, 2001.

Floyd, T., Digital Fundamentals, 8th edn, Prentice Hall, 2003.

CIUJ: UI 4112 Object Oriented Programming

SUT: HIT1151 Software Development 1A

Credit Points: 12.5

Contact Hours: 57 Hours

Duration: 1 Semester

Campus: CIUJ

Prerequisites: Nil

Corequisites: Nil

Teaching Method: Lecture, Tutorial, Laboratory

Assessment: Assignments, Examination

Aims & Objectives

*To demonstrate understanding of the main features of the software development process in an object-oriented framework and explain basic concepts

* To produce simple object-oriented designs from a written specification.

* To write object-oriented programs using Java.

* To co-operately develop and criticize object-oriented designs.

Content

* The object-oriented world view.

* Introduction to object-modelling, implementation of objects and classes.

* Contracts: pre- and post-conditions and assertions.

* Control structures, Input–output, Event-driven programs.

* Introduction to class libraries.

* Use of an OO notation.

Reading Materials

Textbook

Savitch, W., Java: An Introduction to Computer Science and Programming, 3rd edn, Prentice Hall, 2004.

References

Henderson-Sellers, B., A Book of Object-Oriented Knowledge, 2nd edn, Prentice-Hall, Englewood Cliffs, NJ, 1997.

Horstmann, C., Computing Concepts with Java 2 Essentials, 3rd edn, Wiley, 2003.

Lewis, J., Loftus, W., Java Software Solutions, Addison-Wesley, 3rd edn, 2004.

CIUJ:UI4113 Database 1

SUT: HIT2016 Database 1

Credit Points: 12.5

Contact Hours: 46 Hours

Duration: 1 Semester

Campus: CIUJ

Prerequisites: Nil

Corequisites: Nil

Teaching Method: Lectures, Tutorials, Laboratory

Assessment: Assignments, Examinations

Aims & Objectives

- * To provide a solid theoretical foundation to the fundamentals of database design and database systems development.
- * To provide sufficient practical exposure to designing and using database so as to equip students for basic database tasks in industry and government.
- * To provide students with experience in the analysis, design and generation of a simple inquiry and update system, using ORACLE.
- * To give students an understanding of the problem in its context, the need for adequate documentation of the system and management of this data to ensure that the information produced is relevant, accurate and maintainable. Students will use conceptual data analysis methods to produce a logical data model.

Content

- * Information in the organisation.
- * The relational data model.
- * Structured Query Language (SQL).
- * Functional dependency diagrams.
- * Entity relationship analysis.
- * Client server database technologies.
- * Normalisation of data.
- * DBMS terminology and concepts.
- * Data integrity.

Reading Materials

Mannino, M., Database Application Development and Design, McGraw-Hill, 2001.
Benyon, D., Information and Data Modelling, McGraw-Hill Publishing Co. Inc., 1997.
Rob, P., Coronel, C., Database Systems: Design, Implementation, and Management, 5th edn, Course Technology, Cambridge MA, 2002.

CIUJ: UI4121 Introduction to Software Engineering

SUT: HIT1031 Introduction to Software Engineering

Credit Points: 12.5

Contact Hours: 36 Hours

Duration: 1 Semester

Campus: CIUJ

Prerequisites: UI4111

Teaching Method: Lectures (2 Hours per Week), Tutorials (1 Hour per Week)

Assessment: Assignments, Examinations

Aims & Objectives

- * To introduce in a small team environment the basic problems that are encountered in the development of software.
- * To examine some of the current techniques and tools which are used by industry to address the above problems.
- * To allow students to experience the preparation of systems development documentation, working as members of small teams (3–4 persons) and using an object-oriented development perspective.

Content

- * What is software engineering?
- * The software development lifecycle.
- * Techniques for requirements elicitation.
- * Software design as an incremental, iterative process.
- * Software defect management, including defect identification and fault detection.
- * Software validation and verification.

Reading Materials

Bruegge, B., Dutoit, A.H., Object-Oriented Software Engineering, PH, 2000.
Quatrani, T., Visual Modelling with Rational Rose 2000 and UML, A-Wesley, 2000.
Oestereich, B., Developing Software with UML: Object Oriented Analysis and Design in Practice, Addison-Wesley, 1999.
Pfleeger, S.L., Software Engineering: Theory and Practice, 2nd Ed, PH, 2001.
Pressman, R.S., Software Engineering: A Practitioners Approach, 5th Ed, McGraw Hill, 2001.
Fowler, M., Scott, K., UML Distilled, 2nd edn, Addison-Wesley, 2000.
Sommerville, I., Software Engineering, 6th edn, Addison-Wesley, 2001.

CIUJ: 4122Java Programming

SUT:HIT1152 Software Development 2A .

Credit Points: 12.5

Contact Hours: 48 Hours

Duration: 1 Semester

Campus: CIUJ

Prerequisites: UI4112

Corequisites: Nil

Teaching Method: Lectures (2 Hours per Week), Laboratory/Tutorial (2 Hours per Week).

Assessment: Assignments, Examination.

Aims & Objectives

- * To extend and strengthen basic concepts of object-oriented programming using Java.
- * To study GUI software development using Java.

Content

- * Intermediate programming.
- * The dynamic model.
- * Java language and Java system.
- * Graphical User Interface programming in Java.
- * Exceptions.
- * Files and streams.
- * Design principles and introduction to patterns.

Reading Materials

Allen, R.K., Software Development 2, 6th edn. Swinburne, 2004.

Allen, R.K., Bluff, K., Oppenheim, A., Software Development 1, 7th edn. Swinburne, 2004.

Savitch, W., Java: An Introduction to Computer Science and Programming, 3rd Edition, Prentice Hall, 2004.

Horstmann, C.S., Cornell, G., Core Java 2, Vol. 1: Fundamentals, Sun Microsystems Press Java Series, Prentice-Hall, 2003.

Horstmann, C.S., Computing Concepts with Java 2, Essentials, 2nd edn, Wiley, 2000.

Riel, A.J., Object-Oriented Design Heuristics, Addison-Wesley, Reading, MS, 1996.

CIUJ:UI4123 Electronic Systems

SUT:HIT182 Electronic Systems

Credit Points: 12.5

Contact Hours: 5 Hours per Week

Duration: 1 Semester

Campus: CIUJ

Prerequisites: Nil

Corequisites: Nil

Teaching Method: Lectures, Laboratory Work and Tutorials

Assessment: Assignments, Computer-Managed Learning, Examinations, Labs, Lab Reports, Tutorials

Aims & Objectives

This subject provides a basic introduction to analog and digital electronics (including analog DC circuit theory, digital logic and digital electronics, analog AC circuit theory, and amplification). The subject also provides a basic introduction to electromagnetism (including electric and magnetic fields, and the generation of electricity). The subject is structured around 'real-life' examples.

Content

Analog DC electronics: charge, current, voltage, Ohm's law, Kirchoffs laws series and parallel ccts., voltage divider, current divider, simplifying resistor networks, power and power transfer.

Analog AC electronics: alternating current and voltage, frequency, period, phase, amplitude: P-P, Peak, RMS; capacitor circuit and reactance, inductor circuit and reactance, RC, RL circuits (series and parallel), phasor notation; impedance, admittance frequency response of LandC, resonance; ideal transformers.

Amplification: ideal opamp, model, Open loop gain; inverting and non-inverting configuration.

Digital electronics: introduction, digital logic, number systems; boolean operators and truth tables; design and simplification of circuits; boolean laws and identities, S of P representation; K maps; combinatorial logic.

Electromagnetism: electric and magnetic fields, static and changing, magnets, magnetic induction AC generators.

Reading Materials

Hambley, AR, Electrical Engineering, Principles & Applications, Prentice-Hall, 1997.
Serway, RA, Principles of Physics, Saunders College Press, 2nd edn, 1994.

CIUJ: UI4211 Introduction to Human-Computer Interaction

SUT: HIT2024 Introduction to Human-Computer Interaction

Credit Points: 12.5

Contact Hours: 36 Hours

Duration: 1 Semester

Campus: CIUJ

Prerequisites: Introductory Programming, e.g. UE2111

Corequisites: Nil

Teaching Method: Lecture (2 Hours per Week), Laboratory/Tutorial (1 Hour per Week)

Assessment: Assignments, Examination

Aims & Objectives

By the end of the subject, students will be able to:

- * Characterise the basic components of human-computer interaction.
- * Demonstrate a knowledge and understanding of a user-centred approach to interface design.
- * Select, design and conduct appropriate and ethical evaluation protocols and critically evaluate the results.
- * Produce written reports in a standard format and effectively present information in an oral presentation.

Content

- * The nature of HCI.
- * Models of human behaviour: Attention, memory, perception, communication and thinking.
- * User-centred principles in the software development process.
- * User needs and task analysis techniques.
- * Conceptual design and metaphors.
- * Physical design principles, design guidelines and rules.
- * Prototyping techniques.
- * Input–output devices and dialogue techniques.
- * Principles of human-centred software evaluation.
- * Coherence, contextual and participatory design.
- * Evaluation without users: Heuristic evaluations, walkthroughs, automated critiques and predictive models (GOMS, Keystroke Level Model).
- * Evaluation with users: Usability testing, interviews, questionnaires, ethics of user testing.
- * Experimental design and data analysis.

Reading Materials

Preece, J., Rogers, Y., Sharp, H., Interaction Design: Beyond Human-Computer Interaction, John Wiley, New York, 2002.

CIUJ: UI 4212 Software Project Management

SUT: HIT2056 Software Project Management

Credit Points: 12.5

Contact Hours: 3 Hours per Week

Duration: 1 Semester

Campus: CIUJ

Prerequisites: UI 4121

Teaching Method: Lectures, Tutorials

Assessment: Assignments, Examination

Aims & Objectives

This subject builds on the work done in UI4121 on small software development projects, introducing some of the problems of large software development projects. The objectives are:

- * To introduce the concepts and techniques required to direct and control the development of medium- to large-scale software, including project management, quality assurance, software process improvement and software metrics.
- * To broaden students' understanding of possible software development paradigms.

Content

- * Software Project Management, including costing, scheduling and risk management.
- * Introduction to Software Quality Assurance.
- * Introduction to Software Process Improvement.
- * Introduction to Software Metrics.

Textbook

Marchewka, J., Information Technology Project Management: Providing Measurable Organizational Value, John Wiley & Sons, 2003

References

- Pfleeger, S., Software Engineering: Theory and Practice, 2nd Ed, PH, 2002.
Pressman, R., Software Engineering: a Practitioner's Approach, 5th edn, MGH, 2001.
Schwalbe, K., Information Technology Project Management, 2nd Ed, Course Technology, 2002.
Sommerville, I., Software Engineering, 6th edn, Addison-Wesley, 2001.
Humphrey, W., Introduction to the Team Software Process, Addison Wesley, 2000.

CIUJ:UI4213 Data Structures and Algorithms

SUT:HIT2253 Data Structures and Algorithms

Credit Points: 12.5

Contact Hours: 48 Hours

Duration: 1 Semester

Campus: CIUJ

Prerequisites: UI4124

Corequisites: Nil

Teaching Method: Lectures, Laboratories

Assessment: Assignments, Examinations

Aims & Objectives

* To understand and use the common data structures and algorithms.

* To improve skills in Java programming.

Content

* Time analysis of algorithms.

* Collection classes in general.

* Linked lists, iterators, stacks, queues, trees, graphs, has tables.

* Searching and sorting algorithms.

* Relevant facilities in the Java Foundation Class Library.

Reading Materials

Preiss, B.R., Data Structures and Algorithms with Object-Oriented Design Patterns in Java, John Wiley & Sons, 1999.

Budd, T., Classic Data Structures in Java, Addison Wesley, 2001.

CIUJ: UI4221 Operating Systems (Linux).

SUT: HIT2114 Operating Systems (Linux).

Credit Points: 12.5

Contact Hours: 36 Hours

Duration: 1 Semester

Campus: CIUJ

Prerequisites: UI4124 and UI4122 and UI4111

Corequisites: Nil.

Teaching Method: Lecture (2 Hours per Week), Tutorial/Laboratory (1 Hour per Week)

Assessment: Assignments, Examinations

Aims & Objectives: To introduce the fundamentals of operating systems.

Content

- * An overview of computer systems and future trends.
- * Processes and concurrency.
- * Processor scheduling.
- * Memory management and virtual memory design issues.
- * I/O systems and file management.
- * Case studies; UNIX, Microsoft Windows.

Reading Materials

Stallings, W., Operating Systems, 4th edn, Prentice-Hall, 2001.

CIUJ: UI4222 C++ for Java Programmers

SUT: HIT3054 C++ for Java Programmers

Credit Points: 12.5

Contact Hours: 40 Hours

Duration: 1 Semester

Campus: CIUJ

Prerequisites: UI4124

Corequisites: Nil.

Teaching Method: Lecture (2 Hrs per Session), Laboratory (2 Hrs per Session)

Assessment: Assignments, Examination

Aims & Objectives

- * To compare and contrast the features of the C++ programming language with those of Java.
- * To reinforce understanding of Java features by learning their equivalent in the 'parent' C++ language.
- * To introduce the capabilities of C++ as a hybrid, procedural and object-oriented language.
- * To explore in depth the facilities offered by C++ for object-oriented programming.
- * To explore the freedom, efficiency and flexibility C++ offers programmers.
- * To emphasise the defensive programming style required by the C/C++ programming language.

Content

- * C++ as a hybrid programming language: structure of C++ programs, compilation process.
- * Data types: control structures, functions, scoping.
- * Pointers, references.
- * C/C++ strings, namespace, C++ use of 'const'.
- * C++ classes and data abstraction: separating interface and implementation.
- * Stream input–output (standard I/O, device and file I/O).
- * Inheritance, abstract classes, multiple inheritance.
- * Operator overloading, friend function and friend classes, static class members.
- * Polymorphism and late binding, the Vtable.
- * C++ type conversion, RTTI.
- * Exception handling.
- * Function templates and class templates, the STL.

Reading Materials

- Budd, T., C++ for Java Programmers, Addison Wesley, 1999.
Weiss, M.A., C++ for Java Programmers, Pearson International Edition, 2004.
Lippman, S.B., Jajoie, J., C++ Primer, 3rd edn, Addison Wesley, 1998.
Eckel, B., Thinking in C++, 2nd edn, Prentice Hall, 2000.

CIUJ:UI 4223 Advanced Web Development
SUT: HIT3041 Advanced Web Development

Credit Points: 12.5

Contact Hours: 36 Hours

Duration: 1 Semester

Campus:CIUJ

Prerequisites: UI 4124

Corequisites: Nil.

Teaching Method: Lectures (2 Hrs per Week), Laboratory (1 Hr per Week)

Assessment: Assignments, Examination

Aims & Objectives

To introduce the technologies, concepts and techniques associated with the development of World Wide Web systems.

Content

- * Introduction to the World Wide Web: definition, history and fundamental concepts.
- * HTML: document structure, images, links, image maps, tables, frames.
- * Protocols and server technology: HTTP, MIME, URIs.
- * JavaScript: syntax, DOM, forms processing, common tasks.
- * Style sheets: CSS formatting, CSS positioning.
- * DHTML: dynamic techniques.
- * Web design and usability: principles of navigation, usability, style guides.
- * CGI programming: CGI concepts, forms, programming with Perl.
- * XML: syntax, DTDs, XSL.
- * ASP and VB Script: fundamental purpose and operation.

Reading Materials

Textbook

Reding, E., Vodnik, S., HTML Illustrated Complete, 2nd edn, ITP, 2001

References

Dietel, H.M., Dietel, P.J., and Nieto, T.R., Internet & World Wid Web: How to Program, Prentice Hall, 2000.

Castro, E., HTML 4 for the World Wide Web, Peachpit Press, 2000.

Teague J.C., DHTML and CSS for the World Wide Web, 2nd edn, Peachpit Press, 2001.

Negrino, T., Smith, D., JavaScript for the World Wide Web, Peachpit Press, 4th edn, 2001.