The Hong Kong University of Science and Technology

UG Course Syllabus

[Course Title] Modern Compiler Construction

[Course Code] COMP4121

[No. of Credits] 3-credit

[Any pre-/co-requisites] Pre-requisite: COMP 3021 OR COMP 3031

Name: [Instructor(s) Name] Dr. Lionel Parreaux

Email: [Your Email Address] parreaux@cse.ust.hk

Course Description

Compiler implementation techniques are relevant to a wide array of software engineering areas. From parsing ad-hoc configuration file formats to validating complex specification languages to generating efficient code solving data-intensive problems at scale, many important problems of today and tomorrow require knowledge of basic compiler technology. The course exposes students to the essentials of modern compiler construction, including parsing, semantic analysis, program transformation, and code generation. Students will learn to design and implement their own programming language and extend it with a feature of their choice in a small team project. The course focuses on achieving these goals through effective high-level programming techniques, whose mastery will also make students better programmers in general.

Tentative course structure: (optional)

1. Overview, source languages and run-time models
2. Review of formal languages
3. Lexical analysis
4. Syntactic analysis (parsing)
5. Name analysis
6. Type checking
7. Type inference
8. Code generation 1
9. Code generation 2
10. Optimization

Assessments:

[List specific assessed tasks, exams, quizzes, their weightage]

Assessment Task	Contribution to Overall Course grade (%)
Projects	60% (five minor projects worth 5% each, and one major project worth 35%)
Exam	40%
Total	100%

Required Texts and Materials

Alfred V. Aho, Monica S. Lam, Ravi Sethi, Jeffrey D. Ullman: Compilers: Principles, Techniques, and Tools (2nd Edition, 2006)
Torben Mogensen, Basics of Compiler Design, (2010 edition, http://hjemmesider.diku.dk/~torbenm/Basics/)

[Optional] Additional Resources

None