COURSE DETAIL

COMPILER DESIGN I

Country
Ireland

Host Institution
Trinity College Dublin

Program(s)
Trinity College Dublin

UCEAP Course Level
Upper Division

UCEAP Subject Area(s)
Computer Science

UCEAP Course Number
150

UCEAP Course Suffix

UCEAP Official Title
COMPILER DESIGN I

UCEAP Transcript Title
COMPILER DESIGN 1

UCEAP Quarter Units
5.00

UCEAP Semester Units
3.30

Course Description

This course teaches students to define the phases of a typical compiler, including the front and back end. Students learn to identify tokens of a typical high level programming language define regular expressions for tokens and design implement a lexical analyzer using a typical scanner generator. The course explains the role of a parser in a compiler and relate the yield of a parse tree to a grammar derivation design and implement a parser using a typical parser generator, and how to apply an algorithm for a top down or a bottom up parser construction construct a parser for a small context free grammar. The course describes the role of a semantic analyzer and type checking create a syntax directed definition and an annotated parse tree describe the purpose of a syntax tree. The course focuses on the role of different types of runtime environments and memory organization for implementation of typical programming languages. The course describes the purpose of translating to intermediate code in the compilation process. Students design and implement an intermediate code generator based on given code patterns.

Language(s) of Instruction
English

Host Institution Course Number
CSU33071

Host Institution Course Title
COMPILER DESIGN I

Host Institution Campus

Host Institution Faculty

Host Institution Degree

Host Institution Department
Computer Science

Print