LECTURE 7

INTRODUCTION TO COMPILERS

SUBJECTS

Natural languages

- Lexemes or lexical entities
- Syntax and semantics

Computer languages

- Lexical analysis
- Syntax analysis
- Semantic analysis

Compilers

- Compiler's basic requirements
- Compilation process

NATURAL LANGUAGES BASICS

In a (natural) language:

- A sentence is a sequence of words
- A word (also called lexeme of lexical unit) is a sequence of characters (possibly a single one)

The set of characters used in a language is finite (known as the alphabet)

The set of possible sentences in a language is infinite

A dictionary lists all the words (lexemes) of a language

- The words are classified into different lexical categories: verb, noun, pronoun, preposition….

NATURAL LANGUAGES BASICS

A grammar (also considered the set of syntax rules) determines which sequences of words are well formed

- Sequences must have a structure that obeys the grammatical rules

Well formed sentences, usually have a <u>meaning</u> that humans understand

- We are trying to teach our natural languages to machines

With mixed results!!

ANALYSIS OF SENTENCES

<u>Lexical Analysis:</u> identification of words made up of characters

- Words are classified into several categories: articles, nouns, verbs, adjectives, prepositions, pronouns…

<u>Syntax analysis:</u> rules for combining words to form sentences

<u>Analysis of meaning:</u> difficult to formalize

- Easily done by humans
- Gives machines a hard time (although natural language processing is evolving)
    - Big research field for those interested in graduate studies…

COMPUTER LANGUAGE PROCESSING

In computer (or programming) languages, one speaks about a program (corresponding to a long sentence or paragraph)

- Sequence of lexical units or lexemes
- Lexical units are sequences of characters

<u>Lexical rules</u> of the language determine what the valid lexical units of the language are

- There are various **lexical categories**: identifier, number, character string, operator…
- Lexical categories are also known as **tokens**

COMPUTER LANGUAGE PROCESSING

<u>Syntax rules</u> of the language determine what sequences of lexemes are well-formed programs

Meaning of a well-formed program is also called its <u>semantics</u>

- A program can be well-formed, but its statements are nonsensical
- Example:

```
int x = 0;
x = 1;
x = 0;
```

- Syntactically, the above code is valid, but what does it mean??

COMPUTER LANGUAGE PROCESSING

uOttawa
L'Université canadienne
Canada's university

Compilers should catch and complain about lexical and syntax errors

Compilers might complain about common semantic errors:

```
public boolean test (int x){
    boolean result;
    if (x > 100)
        result = true;
    return result;
}
```

Error message:
The local variable result may have not been initialized

Your coworkers or the client will complain about the rest!!

COMPILERS

What is a compiler?

- Program that translates an executable program in one language into an executable program in another language
- We expect the program produced by the compiler to be better, in some way, than the original

What is an interpreter?

- Program that reads an executable program and produces the results of running that program

We will focus on compilers in this course (although many of the concepts apply to both)

BASIC REQUIREMENTS FOR COMPILERS

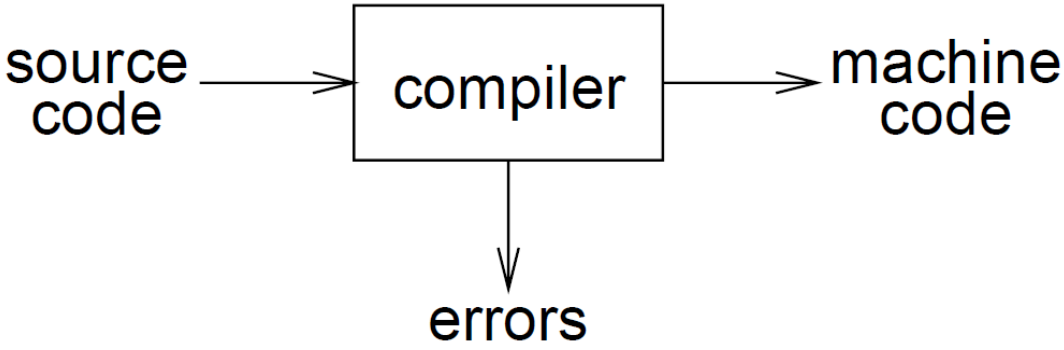
Must-Dos:

- Produce correct code (byte code in the case of Java)
- Run fast
- Output must run fast
- Achieve a compile time proportional to the size of the program
- Work well with debuggers *(absolute must)*

Must-Haves:

- Good diagnostics for lexical and syntax errors
- Support for cross language calls *(checkout Java Native Interface if you are interested)*

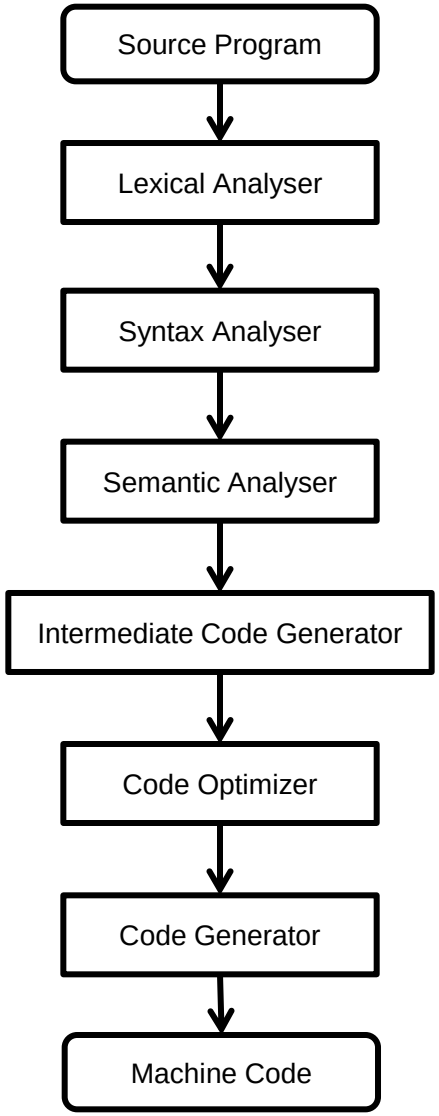
ABSTRACT VIEW OF COMPILERS



A compiler usually realizes the translation in several steps; correspondingly, it contains several components

Usually, a compiler includes (at least) separate components for verifying the lexical and syntax rules

COMPILATION PROCESS



COMPILATION PROCESS

A whole course is required to cover the details of the various phases

In this course, we will scratch the surface

- We will focus on lexical and syntax analysis

SOME IMPORTANT DEFINITIONS

These definitions, although *sleep inducing*, are important in order to understand the concepts that will be introduced in the next lectures

So here we go…

ALPHABET

<u>Recall from beginning of the lecture (or kindergarten):</u> an alphabet is the set of characters that can be used to form a word

…

Since mathematicians love fancy Greek symbols, we will refer to an alphabet as Σ

ALPHABET

Σ **is an alphabet, or set of characters**

- Finite set and consists of all the input characters or symbols that can be arranged to form sentences in the language

English: A to Z, punctuation and space symbols

Programming language: usually some well-defined computer set such as ASCII

STRINGS OF AN ALPHABET

Σ={a,b,c,d}

Possible strings from Σ include

- aaa
- aabbccdd
- d
- cba
- abab
- cccccccccacccc
- Although this is fun, I think you get the idea…

FORMAL LANGUAGES

Σ: alphabet, it is a finite set consisting of all input characters or symbols

Σ^*: closure of the alphabet, the set of all possible strings in Σ, including the empty string ε

A (formal) language is some specified subset of Σ^*

THANK YOU!

QUESTIONS?