INTRODUCTION

CS464 - Compiler Construction

M. Ramzan Shahid Khan
Email: ramzan.shahid@namal.edu.pk

Course Goals

- The areas covered in this course include: Compiler techniques and methodology. Phases of compiler
- Parsing techniques. Contrast b/w compiler and interpreter.

Pre- requisite

- Strong programming background in C, C++
- Formal Language (REs,NFAs, DFAs, CFG)

Textbooks and Class Materials

- ***Compilers: Principles, Techniques and Tools***, Alfred v.Aho, Ravi Sethi , Jeffrey D.Ullman.
- ***Compiler Construction principles and Practice,***

  Kenneth G.Louden
- ***Compiler Design***

  O.G.Kakde
- ***Introduction to computer theory(second Edition)***

  Daniel I.A.Cohen

Answer These Questions

- What is compiler ?
- Why we need compiler?
- What are the high or low level languages?
- Difference b/w them.

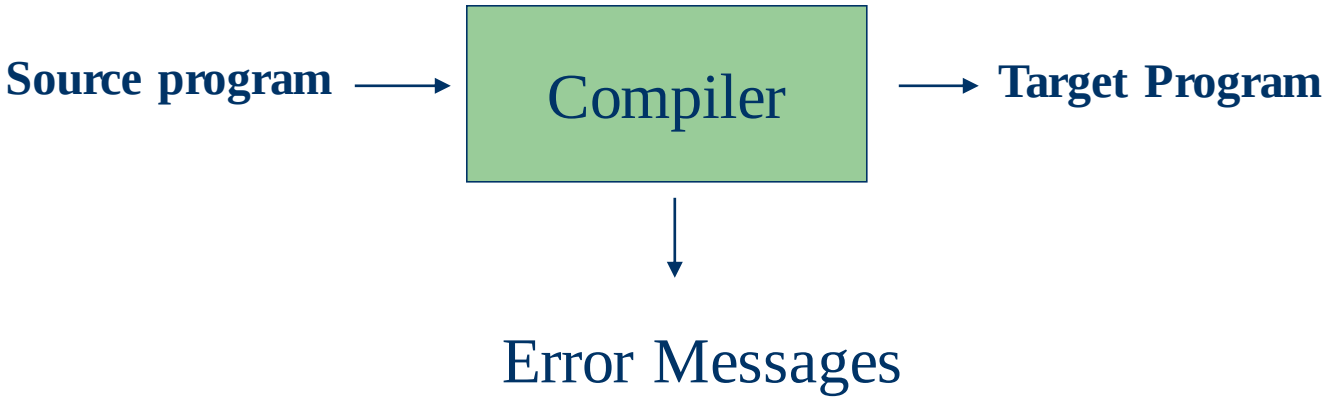
Answer These Questions (cont'd)

- Difference b/w Assembly and Machine language.
- What is the language of computer ?
- Why we feel easy in high level language ?

What is compiler ?

A compiler is a program that reads a program written in one language (i.e source program) And translates it into equivalent program in another Language(i.e the target language).

- The compiler reports to its user the presence of errors in the source program.



Different Terminology

- Compiler are sometimes classified as **SINGLE PASS**,**MULTI PASS** ,**DEBUGGING** depending on how they have been constructed or on what function they are supposed to perform.Despite the apparent complexity, the basic functions that any compiler must perform are essentially the same.

Decompiler

"A program that translates from low level to higher level one is a decompiler "

It can be used for the recovery of lost source code and is also useful in some cases for error correction. In case if the vendor of a software lose its source code ,they can use a decompiler to recover the source code from the executable.

Decompiler (cont'd)

If some error creeps in the executable of a software ,it can be decompiled to make corrections.Example of decompiler is:

JAD- the fast java decompiler –jad is a 100% pure c++ program and claims to be several times faster than decompilers written in java.Jad is free for non-commercial use,but since the version 1.5.6 it's no longer free for commercial use.

The Ethics of Decompilation

- Is de-compilation legal, and is it allowed?

There are many situations when de-compilation can be used...

1) **To recover lost source code**. You may have written a program for which you only have the executable now (or you got the exe of a program you wrote long back, from someone else!). If you want to have the source for such a program, you can use de-compilation to recover it. In all rights, you are the owner of the program, so nobody is going to question you.

The Ethics of Decompilation(cont'd)

2) Just as stated above, **applications written long back for a computer may not have the source code now, and you may need to port it to a new platform**. Either you have to rewrite the application from the scratch, or use de-compilation to understand the working of the application and write it again.

The Ethics of Decompilation(cont'd)

3) Say you have code written in some language for which you cant find a compiler today! **If you have the executable, just decompile it** and rewrite the logic in the language of your choice today

The Ethics of Decompilation(cont'd)

4) To discover the internals of someone else's program (like what algorithm they have used...)

The Ethics of Decompilation(cont'd)

- Usually all software are copyrighted by the authors. This means, copying or expressing the same idea in another program is prohibited.
- Hence if you are using de-compilation to discover the internals of a program and if that particular part is breaking the copyright of the owner, you are liable for legal action. However, there are some permitted uses of de-compilation, like the first three cases stated above

The Ethics of Decompilation(cont'd)

- Also, de-compilation of parts of software which do not come under the copyright laws is permitted. In any case, it is better to contact your legal advisor if you are doing any serious work with de-compilation.

The Ethics of Decompilation(cont'd)

- In all practical purposes, decompiling programs which were created by you can't be questioned! After all, you are the owner of all rights to the program. But be careful if you are trying it out on someone else's programs.

Cross-Compiler

"A compiler that runs on one computer but produces object code for a different type of computer "

Cross compilers are used to generate software that can run on computers with a new architecture or on special-purpose devices that cannot host their own compilers.

Example

- A program developed to control the operations of a microwave oven is an example of such an embedded system.
- As the software environment of a microwave oven is not that much proficient that is can support its own compiler,so a separate compiler is needed to first separately compile the software code.

Example (cont'd)

- After compilation ,the software is embedded into the machine.This separate compiler used to compile the source code is called a cross compiler.

The Analysis-Synthesis Model of Compilation

- There are two parts of compilation **analysis** and **synthesis** .
- The **ANALYSIS** part breaks up the source program into pieces and creates and intermediate representation of the source program

The Analysis-Synthesis Model of Compilation (cont'd)

- The **SYNTHESIS** part constructs the desired target program from the intermediate representation.

The Analysis-Synthesis Model of Compilation (cont'd)

- During analysis the operations implied by the source program are determined and recorded in an hierarchical structure called a **TREE**. Often a special kind of tree called a **SYNTAX TREE** is used ,in which each node represents an operation and the children of a node represent the arguments of the operation.

Example

Position:= initial + rate * 60

Software Tools

- Many software tools that manipulate source programs first perform some kind of analysis.
- Some examples of such tools include.

Structure Editors

- A structure editor takes as input a sequence of commands to build a source program.The structure editor not only performs the text creation and modification functions of an ordinary text editor but it also analyses the program text ,putting appropriate hierarchical structure on the source program. Thus the structure editor can perform additional tasks that are useful in preparation of programs.

Structure Editors (cont'd)

- For example it can check that the input is correctly formed can supply keywords automatically(e.g.when the user types while the editor supplies the matching do and reminds the user that a condition must come between them) and can run from a begin or left parenthesis to its matching end or right parenthesis.
- Further the output of such an editor is often similar to the output of the analysis phase of a compiler.

Static checkers

A static checker reads a program, analysis it and attempts to discover bugs with running the program

For example a static checker may detect the parts of the source program can never be executed or that a certain variable might be used before being defined

Interpreter

Instead of producing a target program as an translation an interpreter performs the operations implied by the source program for an assignment statement, **for example** an interpreter might build a tree and then carry out the operations at the nodes as it walks the tree.

Interpreter (cont'd)

- At the root it would discover it (interpreter) had on assignment to perform so it would call a routine to evaluate the expression on the right and then store the resulting value in the location as associated with the identifier **position**. At the right side of the root the routine would discover, it had to compute the sum of two expressions. It would call itself recursively to compute the value of the expression **rate*60** it would then add that value to the value of the variable **initial.**

THANKS