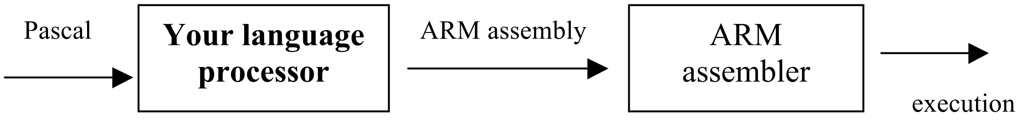
E 2.15 Language Processors

Department of Electrical & Electronic Engineering
Imperial College London

Language Processors Assessed Practical

From Pascal to ARM assembly language

In this practical you are required to design and implement a language processor that converts Pascal source files into their equivalent ARM assembly language source files. Your result ARM assembly code should be executable and demonstrable in the EEE laboratories.



The practical can be broken in the following three parts, of increasing difficulty:

1. **Basic Control Flow and Data structures:** Your processor should be able to handle basic Pascal programs that consist of simple statements containing numeric variables, arithmetic operations, conditional statements, and printing of results **[60%]**
2. **More Advanced Control Flow:** Your processor should be able to handle loops, procedures and functions **[20%]**
3. **More Advanced Data types:** Your processor should be able to handle more elaborate data types, including arrays and pointers. **[20%]**

Your solution should be written in C/C++; you can (*but are not required to*) use the Flex/Lex and Bison/YACC family of tools. The C code generated by these tools can be called from your C/C++ program. Your generated code should be runnable on ARM processors, and you can use the ARMulator, or Keil/µVision (installed on the lab machines) to verify that the generated code is runnable.

Assessment

You are required to submit (through Imperial's VLE BlackBoard) a .zip file containing:

1. A report with a high-level description of your solutions to the above, including grammar choices; you should use this as a medium to describe your design decisions, algorithmic descriptions, discussion on your compiler's behavior/performance, and anything else you want to claim extra marks for. Do not include large pieces of code or grammars here, only small parts if necessary to illustrate a particular point.
2. All your ***well-commented*** code, including (in a README file) any special instructions (e.g. OS used, tools used, and their versions, with corresponding commands) needed to run and test your code.

This report and related files are due at the same time as the oral exam, NOT LATER.

For all parts, the main assessment criteria will be

1. Correctness and quality of generated ARM code (as well as runnability on one of the ARM code development platforms (ARMulator or KEIL/μ Vision).
2. Error checking and reporting.
3. Quality of explanations and supporting documentation.

Extra points will be awarded for any extensions to the above minimum specifications.

The deadline for the deliverables is **Monday, 19 March 2012, at 9am, at the beginning of the last ISE2 lab session**. *In addition to delivering the above*, you will have to attend an oral examination in which you will demonstrate your language processor during the same ISE2 laboratory session (9am-11am), on the 19th of March. **There will be no extensions; absence excusable only with medical note**.

Further information, any frequently asked questions, and a series of representative test programs [a subset of the ones that will be used during the oral examination] appear on the course web page

http://www.iis.ee.ic.ac.uk/yiannis/lp/faq.html