# Compiler Design Homework 5
# Due Before Week 8

**Start Reviewing for Midterm Exam &**
**Prepare a sheet of notes for Mid-Term Exam**

- One (8.5x11) page of notes, (you may use both sides)
- Handwritten in your own hand.
- **To be handed in on Week 7 after (with) the exam**

**Spreadsheet Project (Due Week 8) (Function & Equation)**

- **Add LLVM IR generation** to your spreadsheet to generate a module and store it in the cell's data structure. (You may limit your spreadsheet to a maximum of **8** other cell references in the function interface and equation.)
- Make replacement possible **without memory leaks**. That is, when an equation changes, the old module must be deleted before the one for the new code is stored in the cell.
- Print out the module along with the other parts of the SS cell's data structure.
- Submit a **readme** file, all the **code** you have written for the spread sheet, the **output file** and the **executable** (identify the OS in the readme). Output files should include the SS generated IR. **Submit results in a zip file to Blackboard before the Week 8 class.**

Note that this week's **Compiler Project** homework will come due on **Week 8**, **Week 9** and **Week 10**. The homework is being spread out to allow you to schedule your time around the **Midterm Exam (Week 7)** and the **Spring Break (3/24/2022).**

**Compiler Project (Due Week 8) (Function & Equation)**

- **Add LLVM IR generation** to an ANTLR visitor pattern to generate code to produce an IR module file that can **create IR** for **functions** with code that includes **declarations**, **equations** and **function calls**.
- Use the **C-Input-0.txt** and **generate an IR module file** in the readable format (**.ll**). **Print** out **the IR module** file corresponding to the C-Input-0.txt file.
- Submit a **readme** file, all the **code** you have written for the compiler project, the **executable** (identify the OS in the readme). Submit the **.ll output** files your compiler code generated. **Submit results in a zip file to Blackboard before the Week 8 class.**

**Compiler Project (Due Week 9) ("if" Flow of Control (FoC))**

- **Add LLVM IR generation** to the compiler project to generate an IR module file that can **create IR** for **"if" tests.**
- Use the **C-Input-1.txt** and **C-Input-3.txt** files to **generate IR module files** in the readable format (**.ll**). **Print** out the **IR module file** corresponding to the C-Input-1.txt and C-Input-3.txt files.
- Submit a **readme** file, all the **code** you have written for the compiler project, the **executable** (identify the OS in the readme). Submit the **.ll output** files your compiler code generated. **Submit results in a zip file to Blackboard before the Week 9 class.**

**Compiler Project (Due Week 10) (Globals & "while" FoC)**

- **Add LLVM IR generation** to the compiler project to generate an IR module file that can create **global variables** and **"while" loops**.
- Use the **C-Input-2.txt** and **generate an IR module file** in the readable format (**.ll**). **Print** out **the IR module** file corresponding to the C-Input-2.txt file.
- Submit a **readme** file, all the **code** you have written for the compiler project, the **executable** (identify the OS in the readme). Submit the **.ll output** files your compiler code generated. **Submit results in a zip file to Blackboard before the Week 10 class.**

**Read Louden Chapter 6.3.2–6.3.4 and 6.4–6.4.5 (Symbol Table) for next week**
**Read Kaleidoscope Tutorial Ch. 4 (JIT)**

Optional **Kaleidoscope with ANTLR4** tutorial.