CS46K Programming Languages

Structure and Interpretation of Computer Programs

Term Project Proposal

TERM PROJECT

DR. ERIC CHOU

IEEE SENIOR MEMBER

Motivations

A Term Project is the most essential part of this course. We set up the requirement for a term project in every one of our courses because of the following reasons:

- A course needs to have a core project experience. A courses may go very fast. Therefore, it is impossible to post too many projects. A good term project will help students to put all the knowledge they learn from this course into use.
- Term projects are usually good showcases in job interview activities.
- The starting date for the term project is the first day of the class. The due date is the last day of the class. Students have enough time to work on it.
- Proposal is due on week 10 but you may work on the term project much earlier

Project Ideas

SECTION 1

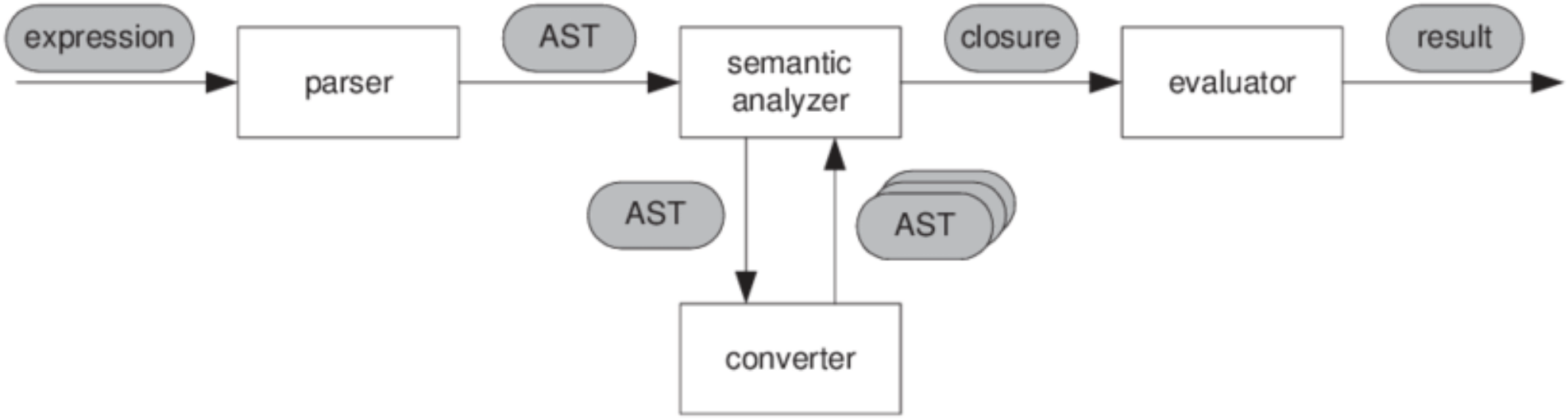
Project Ideas

Any project related to programming language tools will be acceptable. Here are the candidate projects. You may pick anyone from the following (but not limited to).

1. A Scheme Language Interpreter.
2. A single instruction assembler and runner (debugger).
3. GEOJSON Visualizer

Programs developed for other courses or other purposes won't be accepted.

Scheme Interpreter



https://cs61a.org/proj/scheme/

One Instruction Set Architecture Assembler and Runner

Subtract and Branch if Less then or Equal to zero

SUBLEQ a, b, c
Mem[b] := Mem[b] - Mem[a]
if (Mem[b] ≤ 0) goto c

```
main:       inc count
loop:       sble  #1 count done      ; if

            copy  n nC
            mod   #15 nC
            sble  z nC fizzbuzz

            copy  n nC
            mod   #3 nC
            sble  z nC fizz

            copy  n nC
            mod   #5 nC
            sble  z nC buzz

            io::printInt16 n
            jump  nextN
```

OISC

SUBLEQ

```
loop:       sble  hello OUT          ; Outputs char pointed to by hello
            sble  minusOne loop      ; Inc char output ptr to next char
            sble  minusOne checkEnd+1 ; Inc end of string ptr to next char
checkEnd:   sble  z hello HALT       ; Halts program if char at ptr is zero
            sble  z z loop           ; Jumps to loop

;=====================================
;           Data Storage
;=====================================
minusOne:   .word -1                 ; Used to increment ptr
hello:      .asciiz "HELLO, WORLD!\n"
z:          .word 0
lorry@fractal ~/d/d/a/s/t/fixtures (master)>
sblasm helloworld.asq > helloworld.sq
lorry@fractal ~/d/d/a/s/t/fixtures (master)> cat helloworld.sq
16 -1 3 15 0 6 15 10 9 31 16 -1 31 31 0 -1 72 69 76 76 79 44 32 87 79 82 76 68 3
3 10 0 0
lorry@fractal ~/d/d/a/s/t/fixtures (master)> sblrun helloworld.sq
HELLO, WORLD!
```

```
; Outputs "HELLO, WORLD!\n"
.equ        OUT  -1
.equ        HALT -1

loop:       sble  hello OUT        ; Outputs char pointed to by hello
            sble  #-1 loop         ; Increments char output ptr to next char
            sble  #-1 checkEnd+1   ; Increments end of string ptr to next char
checkEnd:   sble  z hello HALT     ; Halts program if char at ptr is zero
            sble  z z loop         ; Jumps to loop

hello:      .asciiz "HELLO, WORLD!\n"
z:          .word   0
```

https://youtu.be/o0e7_U7ZmBM

Geometry primitives

Type	Examples	

Point		{
		"type": "Point",
		"coordinates": [30.0, 10.0]
		}

LineString		{
		"type": "LineString",
		"coordinates": [
		[30.0, 10.0], [10.0, 30.0], [40.0, 40.0]
		]
		}

Polygon		{
		"type": "Polygon",
		"coordinates": [
		[[30.0, 10.0], [40.0, 40.0], [20.0, 40.0], [10.0, 20.0], [30.0, 10.0]]
		]
		}

		{
		"type": "Polygon",
		"coordinates": [
		[[35.0, 10.0], [45.0, 45.0], [15.0, 40.0], [10.0, 20.0], [35.0, 10.0]],
		[[20.0, 30.0], [35.0, 35.0], [30.0, 20.0], [20.0, 30.0]]
		]
		}

GEOJSON Visualizer

https://en.wikipedia.org/wiki/GeoJSON

Schedule

SECTION 2

Schedule

1. The project can start at any time after the course starts.
2. In week 10, you will be required to submit a **100-point** worth project proposal.
3. In week 16, your project will be due, you will be required to submit all your deliverables before the end of the course. Late submission may not be graded. The final project submission is of **700-point** worth.

Project Proposal

Mauris enim leo, rhoncus sed, vestibulum sit amet, cursu

Date of Submission


23% YES
35%
39%

Deliverables

SECTION 3

Deliverables - Proposal

- In week 10, a proposal should be submitted. In the proposal, you should include the following items:
  1. project title and idea
  2. project deliverables
  3. tools
  4. materials
  5. schedule

Deliverables – Final Report

- In week 16, a final report should be submitted.
  1. A final report in .docx or .pdf
  2. A presentation made by the student in
     - a PowerPoint,
     - google slides or
     - a YouTube link.
  3. All the technical files (program files, design schematics, etc.)
- All materials can be grouped into a directory and be compressed to a .zip file. The YouTube video link is good enough. Do NOT send the whole video presentation file over to us.

Grading

SECTION 4

All Grading by the Grader is Final

- Grade is final and not negotiable.
- All grading on project are subjective. It is impossible to stay completely neutral and objective.
- We try to be fair but can never promise to be always accurate.
- Non-the-less, we will try our best to be in favor of students. As long as you demonstrate your efforts, you will get a good grade.

Grading Policy

The total base point for the term project is 700 pts. A maximum of 100 pts may be awarded to you as bonus points.

The final term project will be judged based on the following factors:

1. Originality
2. Creativity
3. Completeness
4. Technical Difficulty
5. Presentation


SPECIAL
BONUS