EE/CprE/SE 492 - sdmay21-10
Dynamic Programming Based FPGA Hosted Control-Flow Integrity (CFI)
Report 3

February 23rd - March 1st
Client: Akhilesh Tyagi
Faculty Advisors: Zelong Li, Ananda Biswas

Team Members

Gregory Wendt - Meeting Scribe
Cole Schumacher - Meeting Facilitator
Nickolas Mitchell - Chief Engineer (FPGA)
Sam Henley - Chief Engineer (Software)
Maxwell Wangler - Test Engineer
Tristan Duyvejonck - Report Manager

Past Weeks Accomplishments

Over the past week our team moved closer to a working FPGA solution. Nick created a finite state machine (FSM) to better translate our software solution to VHDL. Greg and Max researched C-Unit to test the new c code. Sam and Tristan continued to research writing a custom LLVM pass. Cole continued research in different FPGA solutions. Below is some code and Nick's FSM we created over the past week.

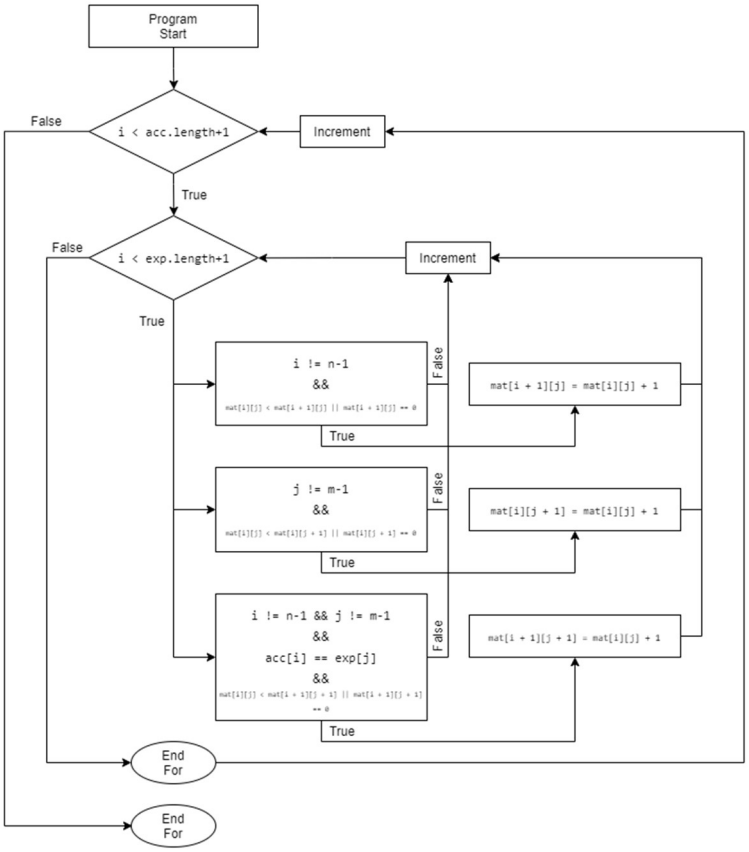

Figure 1: Flowchart of RaceLogic Algorithm (Makeshift FSM) (Credit: Nickolas)

```
unsigned char* list_to_array(struct linked_list* list)
{
    unsigned char* arr = (unsigned char*) malloc((sizeof(char))*list->size);
    struct linked_list* cur = list;
    for(int i = 0; i < list->size; i++)
    {
        arr[i] = cur->value;
        cur = cur->next;
    }
    return arr;
}

struct linked_list* add_to_list(struct linked_list* list, unsigned char new_label)
{
    if(list == NULL)
    {
        list = (struct linked_list*) malloc(sizeof(struct linked_list));
        list->next = NULL;
        list->value = new_label;
        list->size = 1;
    }
    else
    {
        struct linked_list* cur = list;
        while(cur->next != NULL)
        {
            cur = cur->next;
        }
        struct linked_list* new_node = (struct linked_list*) malloc(sizeof(struct linked_list));
        new_node->next = NULL;
        new_node->value = new_label;
        cur->next = new_node;
        list->size++;
    }
    return list;
}

void free_list(struct linked_list* list)
{
    while(list != NULL)
    {
        struct linked_list* next = list->next;
        free(list);
        list = next;
    }
}
```

Figure 2: Code of Input creations (Credit: Maxwell)

Pending Issues

- Work towards a better understanding of LLVM compilers
    - Debug issues with LLVM
    - Figure out how to get the generated labels (compile-time and runtime)
- Finish VHDL solution
    - Test VHDL code
- Look into ways to run programs on our FPGA board
- Test 8-bit string inputs

Individual contributions

NAME	Individual Contributions (Quick list of contributions. This should be short.)	Hours this week	HOURS cumulative
Gregory Wendt	Researched into c unit for c code testing and added clarification comments to old tests	2	10
Cole Schumacher	Research into FPGA implementation, helping coordinate meeting	3	10
Nickolas Mitchell	Completed FSM model for FPGA design	3.5	20
Sam Henley	LLVM custom pass	4	9
Maxwell Wangler	Researched into c unit for c code testing	1.5	8.5
Tristan Duyvejonck	Helped Sam with LLVM tests	2	3

Plans for the Upcoming Report

- Gregory
  - Work with Max to test 8bit instructions
- Cole
  - Help Nick with FSM and VHDL code
- Nickolas
  - Use FSM to continue work on FPGA solution
    - Test VHDL code
  - Look into ways to run programs on our FPGA board
- Sam
  - Continue testing and developing LLVM passes
- Maxwell
  - Continue testing 8bit instructions
- Tristan
  - Look into FPGA board to fit project requirements