Program Assignment 1 Min Heap

Oct 04, 2018

By Jing-Jia Liou

Contents

1 Problems

- For this assignment, you have to write an assembly code. We have prepared a code template for you. Please try to use it to define your data array. You can download it using the following commands (assume you have already done lab0 or lab1 and the directory `~/ee3450` has been created):

```
$ cd ~/ee3450

$ git clone http://gitlab.larc-nthu.net/ee3450_2018/pa1.git
```

- You'll write some RISC-V assembly code to implement a heapify algorithm. There are three parts of this assignment:

  1. Write a RISC-V assembly code to generate min heap structure from an unsorted array.
  2. Prepare an unsorted array as a test case.

1.1 Penalty of plagiarism

- Each time you submit a plagiarized code, your grade of the homework will be discounted to 90%. The check is done in batch every hour. The check is based

on code similarity at a 30% threshold (meaning 30% of your code is structurally identical to other's codes).

- The deduction will be **accumulated each time** you submit a plagiarized code. Please do not use trial-and-error approach to adjust your codes.
- If your final version (last submission before due date) is a plagiarized code, no credit will be given.

1.2 Part I Build a Min Heap

Due: 11:59 am on Oct. 11, 2018

Proportion: 85%

- Heap

  #. A heap must be a complete binary tree, this feature makes it suitable to be stored in an array.

  #. In a Min Heap, the key at root must be minimum among all keys present inside the heap. The same property holds for all the nodes in a Min Heap. For more details, please refer to MinHeapify. You only need to understand the part about min heap concept.
- For this part, your program should build a "min heap" structure from a given array (written in the assembly code directly).
- For example, given an array [8, 7, 15, 4, 20, 10, 2], the output, [2, 7, 4, 8, 15, 20, 10], is one of the possible legal solution.
- Due to different implementation (or processing order), your output may be different. Please make sure your output is a valid Min Heap. We will announce a min heap check program later.
- The template for your assembly code is in `pa1/pa1-1-heapify`, as shown below:

```
.section .data
# There are two global variables:
#  1. array: Store the input numbers.
#  2. size: the number of the array
# You can modify the following two lines to test your program,
# but DO NOT change the line number
```

```
array: .dword NUM1, NUM2, NUM3, NUM4, NUM5, NUM6, NUM7
size: .dword 7

.section .text
.global main
main:

  # your code goes here

  li  a0, 0      #assigne 0 to a0
  ret            #return (jump to the addr store in register ra)
```

- In the template code, `array` is a array that stores 7 numbers. `NUM1, NUM2, NUM3, NUM4, NUM5, NUM6, NUM7` should be replaced by actual numbers, e.g., 8, 7, 15, 4, 20, 10, 2 before compiling. Another variable `SIZE` is the size of the array. You need to copy and paste from test data to fill in these numbers manually. (Our test program will replace them automatically).
- Using following commands to compile

```
$ cd pa1/pa1-1-heapify
$ make
```

- Please refer the lab1 to run and debug your program.

1.3 Part II Test Case

Due: 11:59 am on Oct. 11, 2018

Proportion: 15%

- Your test case should look like this

```
9
8, 7, 15, 4, 20, 10, 22, 12, 2
```

- Rule

1. The first line should be the number of integers. The maximum number is limited by the size of immediate field of RISC-V instructions (2^11-1). Make sure the number represents a valid complete binary tree.
2. The second line includes positive or negative **32-bit integers**. The same integers can appear in the array.
3. The integers in the second line are separated with a comma and space.

1.4 Min Heap Checker

- We have a checker program to validate whether your output array is a min heap or not.
- You have to create a file (or you can refer to lab1: Calling assembly functions in C and use stdout to generate this file)with the same format as your test case file like this

```
4
8, 7, 15, 4
```

- Then use the following command to check if your output is a min heap (assume you have logged in ee3450B/C/D):

```
$ vim test.txt
$ ~ee345000/pa1/bin/check < test.txt
Not a min heap.
$
```

- As the checker shows, the array is not a valid min heap. Try modify test.txt like:

```
4
4, 8, 7, 15
```

- Run checker again.

```
$ vim test.txt
$ ~ee345000/pa1/bin/check < test.txt
Format correct! It is a min heap.
$
```

1.5 Submission

- For example, if your student ID is **103061232**,
  1. Your part I assembly file name will be **hw1_103061232.S**
  2. Your test case file name will be **hw1_103061232.txt**
  3. Submit your home work via the link.

1.6 Hints

- Before writing the assembly code, we highly recommend you to write this program in high-level language first (you may reuse codes in above reference link).