CENG311 **Fall 2021**

PROGRAMMING ASSIGNMENT 2

Due date: 24.11.2021 23:00

In this assignment, you are required to implement some priority queue operations in MIPS assembly language. You will use SPIM simulator [1] to develop and test your code.

First of all, you will implement a max binary heap to represent the priority queue, where the value of parent node will always be greater than the value of child node, and the value at the root must be maximum among all the values. Your heap nodes will have the following structure:

Byte Address:	**Contents**
X:	Value of the node
X + 4:	Address of the left child
X + 8:	Address of the right child
X + 12:	Address of the parent

Each node of your heap should be stored in 16-byte address which holds the value of the node, the address of the left child, the address of the right child, the address of the parent node respectively. When there is no left child or right child or parent, the memory address is 0.

Procedures to be Implemented:

build (*list, queue*)
This procedure will construct a priority queue data structure from an unordered list of integers. The address of the first integer is in the ***list*** argument (assume the list is terminated by a special value, -MAXINT (the negative integer with the largest possible absolute value)), and the address of the place where the procedure should create the data structure in ***queue*** argument (assume there is enough space for the queue structure). This procedure will call ***insert*** procedure for each node insertion operation.

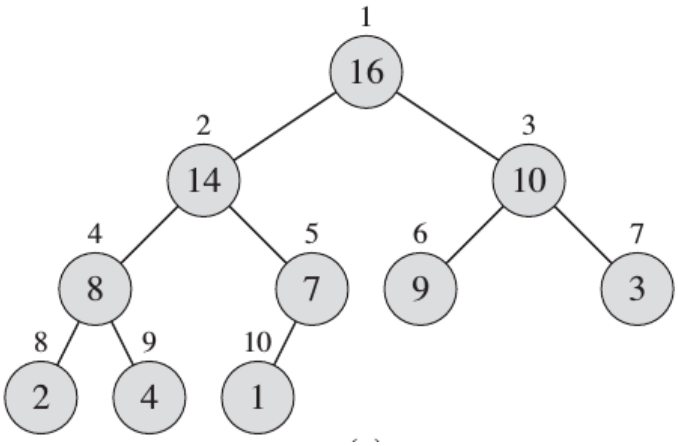
insert (*value, queue*)
This procedure will create and put a new node (the value is given in $a0 register) to the queue (the address of the root node of the queue is in ***queue*** argument). The procedure will require new space in memory for the new node, which can be obtained with the MIPS system call *sbrk*. The address of the location where the new node is inserted should be stored in $v0 register.

remove (*queue*) -OPTIONAL-
This procedure will delete the maximum element in the queue, which is the root node (the address of the root node of the queue is in ***queue*** argument). The removed value should be stored in $v0 register.

print (*queue*)
This procedure will print the priority queue (the address of the root node is in ***queue*** argument) to the screen by breadth first traversal.

You can implement additional procedures if you need.

Example priority queue:



Example output of **print** procedure:
16 14 10 8 7 9 3 2 4 1

Assumptions:

- The arguments to the procedures are stored in $a registers, the first is in $a0, the second is in $a1, and so on.
- Only valid arguments are passed into the procedures. You do net need to check the arguments for their validity.
- The input and output lists have maximum 100 characters-length.

Requirements:

- The values in all $a registers should be as they were at the time of call at the end of the procedure.
- Your code should be ready for a test program (provided by your TA) which checks flow and result of the procedures.
- You are required to submit the source code and a report that includes implementation details and screenshots of your sample executions.
- You need to work individually, no group work is allowed.
- No late homework will be accepted.

Submission: You are required to submit your commented source code and report to cloud-lms. Please create a compressed file including all source files and report; and name it as yourstudentnumber_P2.zip (e.g. If your student number is 202112345678, the file name must be 202112345678_P2.zip).

[1] http://spimsimulator.sourceforge.net/
[2] Data Structures and Algorithm Analysis in C, Mark A. Weiss, Addison Wesley