Reserved Cells

Ex. 1	
Ex. 2	
Ex. 3	
Ex. 4	
Ex. 5	
Ex. 6	
Tot.	

Algorithms and Programming

13 September 2018

Part I: Theory

Register Number _____ Family Name _____ First Name _____

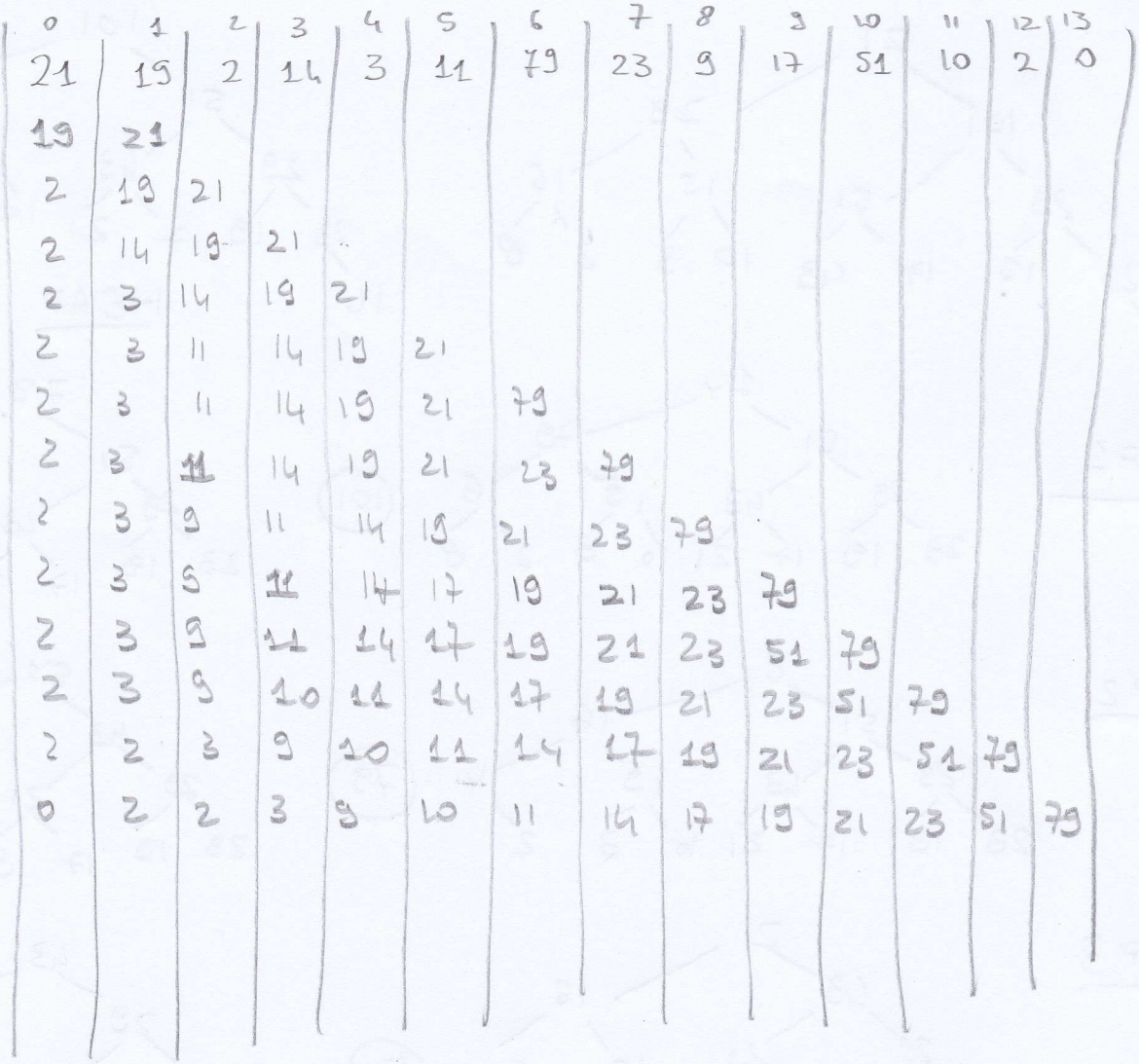
Course: ○ 10 credit course (01*OGDLP*) ○ 12 credit course (02*OGDLM*)

No books or notes are allowed. Solve exercises directly within the reserved space. Additional sheets are accepted only when strictly necessary. Examination time: 50 **minutes.**

1. **(1.0 points)**
   Sort in ascending order with insertion sort the following array of integers:

   21 19 2 14 3 11 79 23 9 17 51 10 2 0

   Show all relevant intermediate steps.

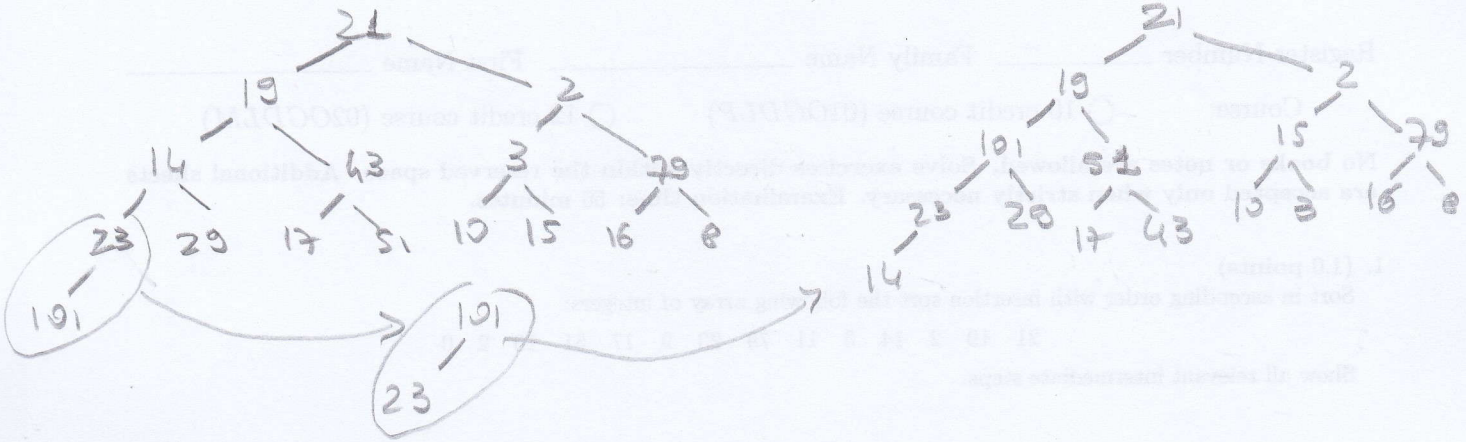


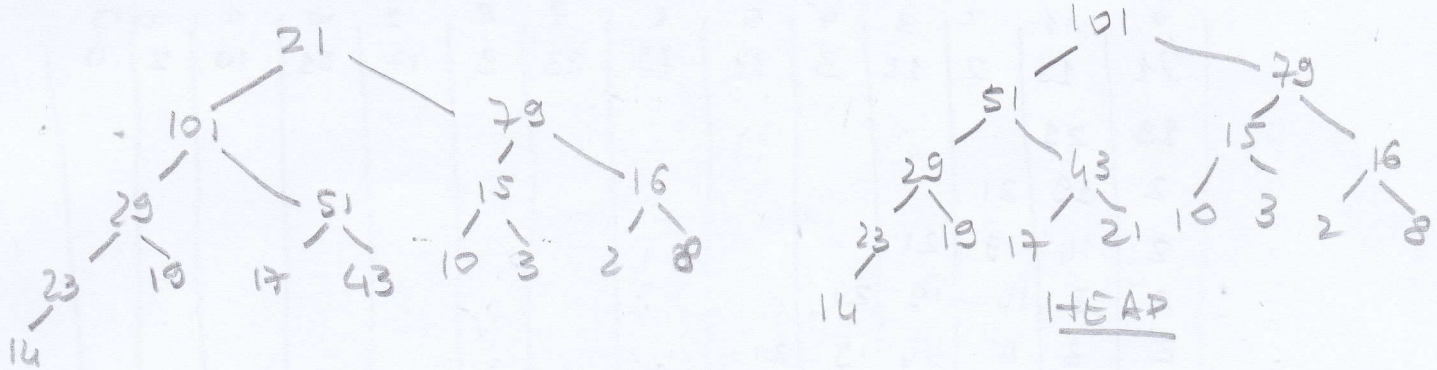
2. (2.0 **points**)
Given the following sequence of integers stored in an array:

21 19 2 14 43 3 79 23 29 17 51 10 15 16 8 101

turn it into a heap, assuming to use an array as underlying data structure. Draw each step of the heap-building process, as well as the final result. Assume that, at the end, the largest value is stored at the heap's root. Execute the first three steps of the heap-sort algorithm on the heap built at the previous step.

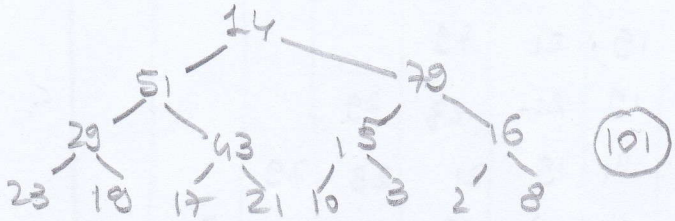
Assume that the sequence is already stored in the array and that it represents an intermediate configuration on which the heap property doesn't necessarily hold.

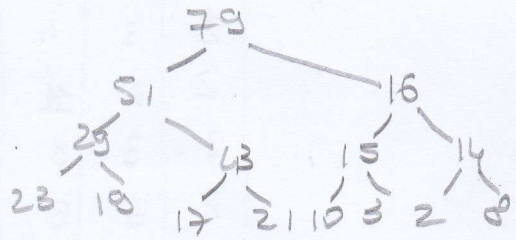




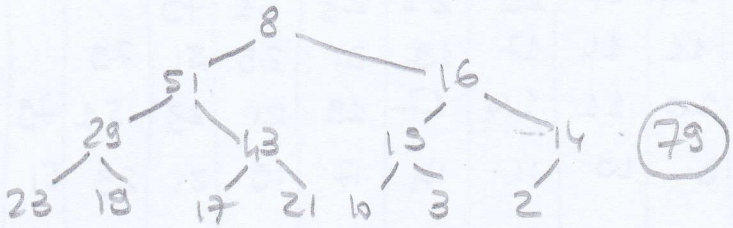
HEAP

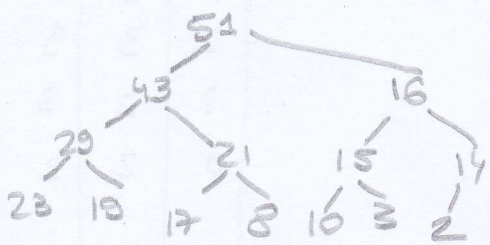
STEP 1



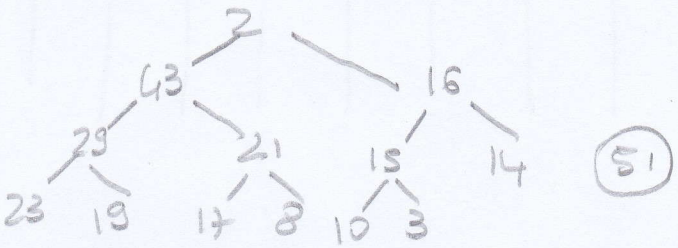


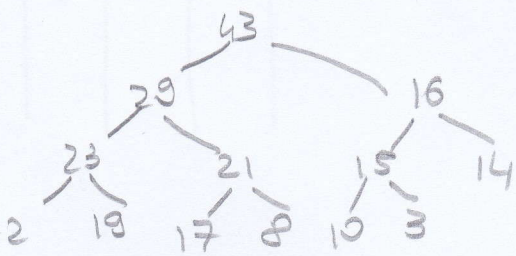
STEP 2





STEP 3





3. **(2.0 points)**

10 credit course (01*OGDLP*)

Consider a binary tree with 11 nodes. Its visits return the following sequences:

pre-order:	21	33	12	7	6	9	10	5	17	13	2
in-order:	12	33	6	9	7	21	5	17	10	2	13
post-order:	12	9	6	7	33	17	5	2	13	10	21

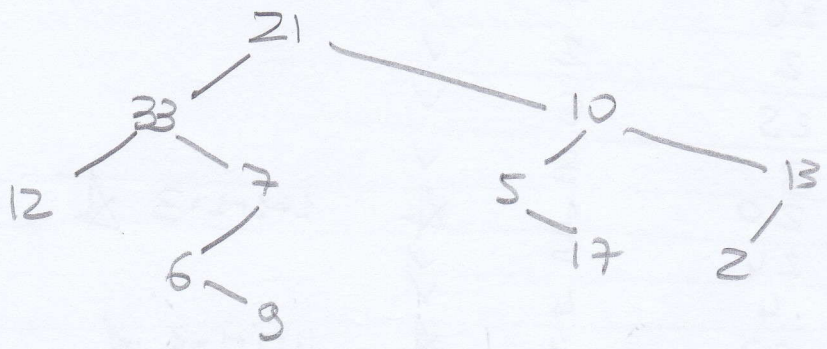
Draw the original binary tree.

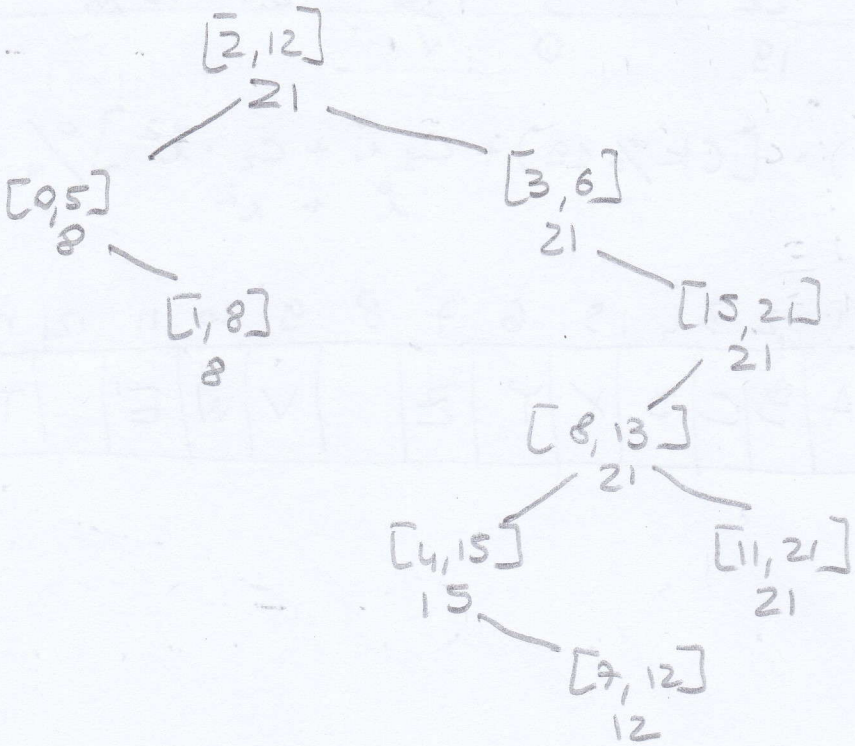
12 credit course (02*OGDLM*)

Insert the following keys in an Interval BST initially empty.

$$[2,12] \quad [3,6] \quad [0,5] \quad [15,21] \quad [8,13] \quad [1,8] \quad [11,21] \quad [4,15] \quad [7,12]$$

Insertions have to be made on the leaves.





4. **(2.5 points)**
Given the sequence of keys

$$A\ Z\ B\ Y\ C\ T\ X\ D\ W\ E\ V\ S$$

where each character is identified by its index in the English alphabet ($A = 1, \ldots, Z = 26$), draw the final configuration of an initially empty hash table of size 19 where insertion of the previous sequence character by character occurs.

Assume open addressing with quadratic probing with $c_1 = 1$ and $c_2 = 1$. Show all relevant intermediate steps.

1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26
A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z

	k	k%19	i=1	i=2	i=3
A	1	1	✓		
Z	26	7	✓		
B	2	2	✓		
Y	25	6	✓		
C	3	3	✓		
T	20	1	✗	1+1+1=3 ✗	1+2+4=7 ✗	1+3+9=13 ✓
X	24	5	✓		
D	4	4	✓		
W	23	4	✗	4+1+1=6 ✗	4+2+4=10 ✓
E	5	5	✗	5+1+1=7 ✗	5+2+4=11 ✓
V	22	3	✗	3+1+1=5 ✗	3+2+4=9 ✓
S	19	0	✓		

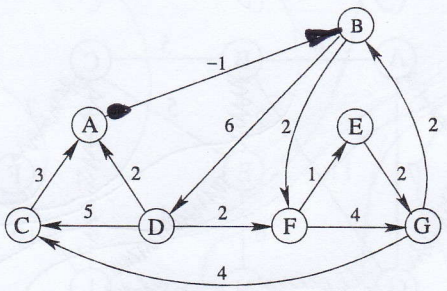
$$h(k) = [(k \,\%\, 19) + c_1 \cdot i^0 + c_2 \cdot i^2] \,\%\, 19$$
$$i + i^2$$

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
S	A	B	C	D	X	Y	Z		V	W	E		T					

5. (2.5 points)

10 credit course (01*OGDLP*)

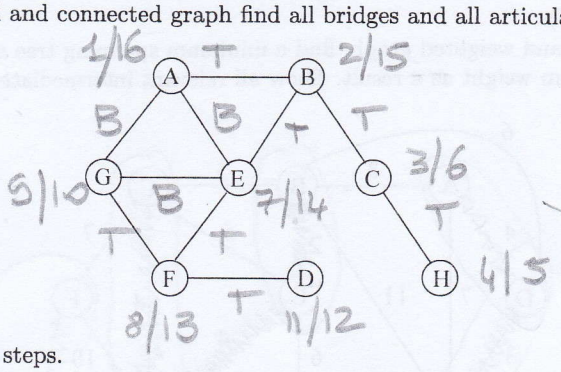
On the following directed and weighted graph, find all shortest paths connecting node A with all the other nodes resorting to Bellman-Ford's algorithm. If necessary, consider nodes in alphabetical order.



AB -1
BD 6
BF 2
CA 3
DA 3
DC 5
DF 2
EG 2
FE 1
FG 4
GB 2
GC 4

12 credit course (02*OGDLM*)

Given the following undirected and connected graph find all bridges and all articulation points.

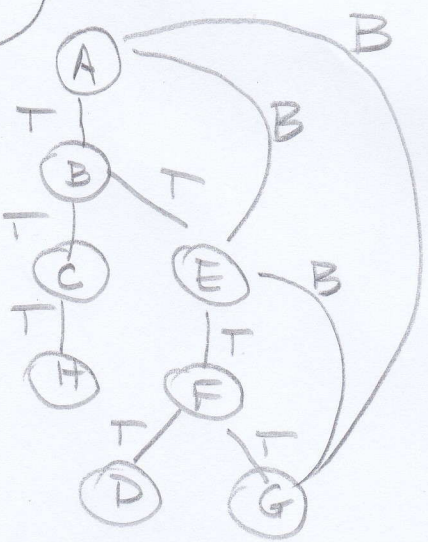


Show all relevant intermediate steps.

	1	2	3	4	5	6	CHECK
A	0	0	0	0			
B	-	-1	-1	-1			
C	-	~~10~~ 9	8	8			
D	-	5	5	5			
E	-	2	2	2			
F	-	1	1	1			
G	-	5	4	4			

STOP ✓

No Negative Cycles



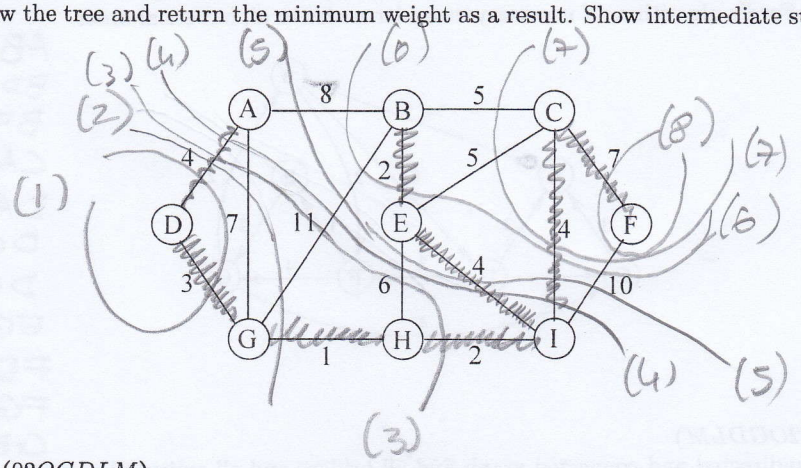
BRIDGES BC
CH
FD

ARTICULATION POINTS B
C
F

6. **(2.0 points)**

10 credit course (01*OGDLP*)

Given the following undirected and weighted graph, find a minimum spanning tree using Prim's algorithm starting from vertex D, draw the tree and return the minimum weight as a result. Show intermediate steps and all generated cuts.

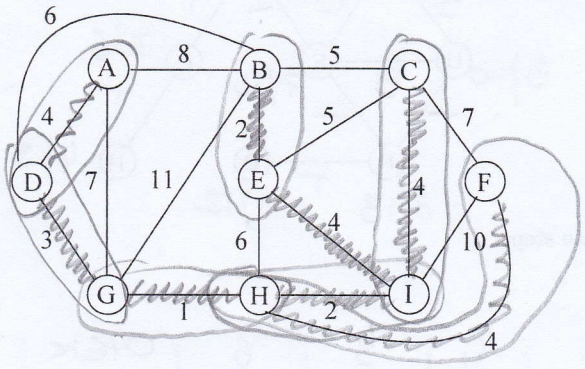


DG 3
GH 1
HI 2
DA 4
EI 4
BE 2
CI 4
CF 7

27

12 credit course (02*OGDLM*)

Given the following undirected and weighted graph, find a minimum spanning tree using Kruskal's algorithm, draw the tree and return the minimum weight as a result. Show all relevant intermediate steps.



GH 1
BE 2
HI 2
DG 3
AD 4
CI 4
EI 4
FH 4

24