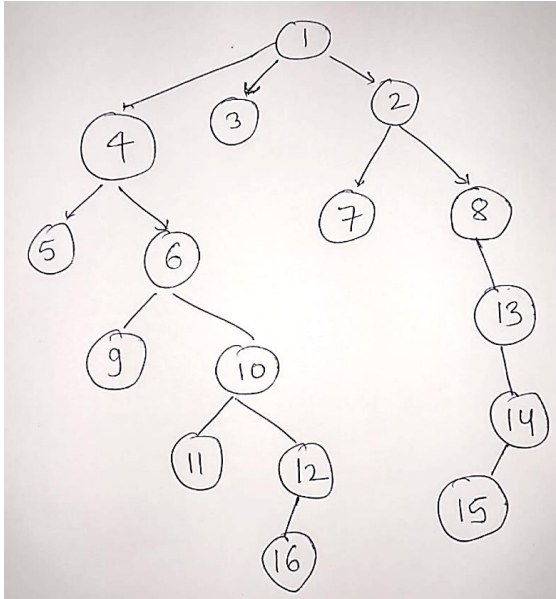


Q1. The famous Hardware software codesign tool Ptolemy uses GCLP algorithm. Consider the following application graph and the parameters associated with each of the nodes. ts_i is the execution time of node i in software, th_i is execution time in hardware, ah_i is the hardware area for node i , and $size_i$ is the size of node i (which is indicative of the number of atomic instructions in the same). HR1 and HR2 are the two hardware repeller properties whereas SR1 and SR2 are the two software repeller properties. [19 marks]



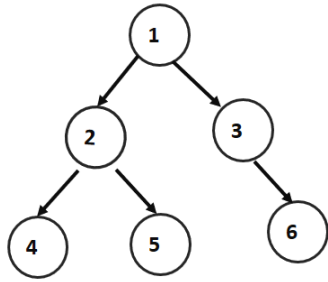
node	ts_i	th_i	ah_i	$size_i$	HR1	HR2	SR1	SR2
1	12	2	15	45	23	0	10	45
2	14	3	30	225	6	1	14	42
3	3	2	23	75	11	0	19	41
4	12	3	90	270	8	1	13	41
5	15	8	75	75	5	1	19	41
6	6	3	180	150	3	0	11	33
7	8	2	75	150	9	1	16	41
8	8	6	120	90	11	0	10	42
9	23	8	225	750	30	1	14	35
10	6	2	15	75	20	1	16	30
11	9	5	180	120	14	0	15	32
12	3	2	23	60	12	0	12	36
13	11	5	45	75	9	1	9	41
14	9	3	30	60	26	1	20	35
15	18	12	60	300	23	0	15	30
16	20	4	70	450	6	0	15	43

- Identify the hardware extremity nodes and the software extremity nodes given that $\alpha = 68.75$ percentile and $\beta = 75$ percentile. [2M + 2M] (hint: i.e. for evaluating software extremity consider candidate nodes in top 31.25%)
- Evaluate the extremity measure for the extremity nodes identified in the previous part [4M + 3M].
- Evaluate the repeller value R_i for the nodes and answer in the format shown below. [8M]

Node	1	2	3	4	5	6	7	8
Repeller measure								

Q2. Consider the application graph shown below with 6 nodes and the parameters associated with the different nodes shown in the table below. Given that the priority function for moving from s/w to h/w is given as $pf = \text{round}(3 ts_i + 2 * (ts_i/th_i) + 120/ah_i)$. Further the time deadline given by the client is 28 ms. Assume that whenever a node is moved to hardware, a dedicated hardware component is added in the system to perform that task. Neglect repeller properties (repeller measure) and extremity measure and assume all nodes to be normal nodes. Assume that initially all nodes are unmapped. Solve the mappings using GCLP algorithm and indicate the ready nodes at each step of the GCLP algorithm. [9M]

Note: Node mapped in this stage & mapping (i.e. h/w or s/w) (NMIS&M)



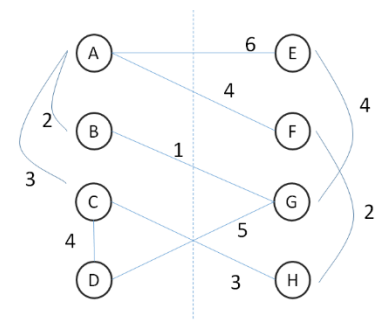
Node	ts_i	th_i	ah_i	size_i
1	6	4	20	260
2	12	8	10	600
3	8	4	30	400
4	10	4	40	300
5	6	4	30	200
6	4	2	16	100

Q3. Consider that you have to solve a mapping problem using simulated annealing. The system has 5 tasks as its subsystems and 5 different boards are available on which the tasks can be mapped. The task allocation constraints are given, and additionally a small set out of the feasible space is given to explore using simulated annealing (design points shown on right). Given that k (Boltzmann constant) = 1.2, $T_{\text{initial}} = 100$ K and $\alpha = 0.7$. Fill up the table given below with appropriate values given that the starting solution is design point 1 and at every step the next solution chosen is Design point (n+1), where n is the current solution design point. Also the random probabilities have already been provided. Fill in the empty entries in the sheet in format as shown below [7M]

Execution time						Design Points										
Task	ARM	ASIC	FPGA	RISC	DSP	Task	1	2	3	4	5	6	7	8	9	10
	τ	τ	τ	τ	τ		ARM	ARM	ARM	ARM	RISC	ARM	RISC	RISC	RISC	RISC
T_1	5	2	3	4	5	T_1	ARM	ARM	ARM	ARM	RISC	ARM	RISC	RISC	RISC	DSP
T_2	13	4	4	12	20	T_2	ASIC	ASIC	ASIC	ASIC	ASIC	ASIC	ASIC	ASIC	ASIC	ASIC
T_3	6	5	10	7	20	T_3	ARM	ARM	ARM	RISC	ARM	RISC	ARM	RISC	RISC	RISC
T_4	16	1	2	20	21	T_4	FPGA	DSP	FPGA	FPGA	FPGA	DSP	FPGA	FPGA	DSP	FPGA
T_5	2	5	6	1	14	T_5	ARM	RISC	RISC	RISC	ARM	ARM	RISC	RISC	ARM	ARM
Task allocation Constraints						Execution Time	19	37	18	19	18	39	17	18	38	20
Task	Component															
	ARM	ASIC	FPGA	RISC	DSP											
T_1	✓			✓	✓											
T_2		✓														
T_3	✓			✓												
T_4		✓	✓		✓											
T_5	✓			✓												

Present soln.	Temperature [1M]	Random prob.	Next soln. [1M]	Acceptance [3M]	prob.	Selected soln. [2M]
1	100	0.3				
		0.5				
		0.2				
		0.5				
		0.6				
		0.8				

Q4. Consider the problem of partitioning nodes in a process into two partitions. The process has 8 functions (nodes) with the cost function between the different nodes indicated on the links shown below. Apply KL algorithm to the following process and show the swapped nodes and evaluate the cutsizes at every step of KL algorithm [6M + 2M]. Evaluate the optimal partitioning after one pass of KL algorithm (i.e. after all the nodes have undergone swapping) and draw in space provided on right [2M]. Assume the initial partition for the KL algorithm as shown on the right side. [10M]



Q5. You are given the task of writing a pseudo code to evaluate the pareto optimal points in the following scenario where you have been given 16 different possible partition configurations (solutions) and 3 parameters are evaluated (Area(**D**), Power (**A**) and Delay (**P**)) for each of them as shown in the table below. Write a pseudo code (in logic language) to evaluate the pareto optimal points among the different configurations. (Note: you just need to write the pseudo code and not do any mathematical calculations for this part). [5 marks]

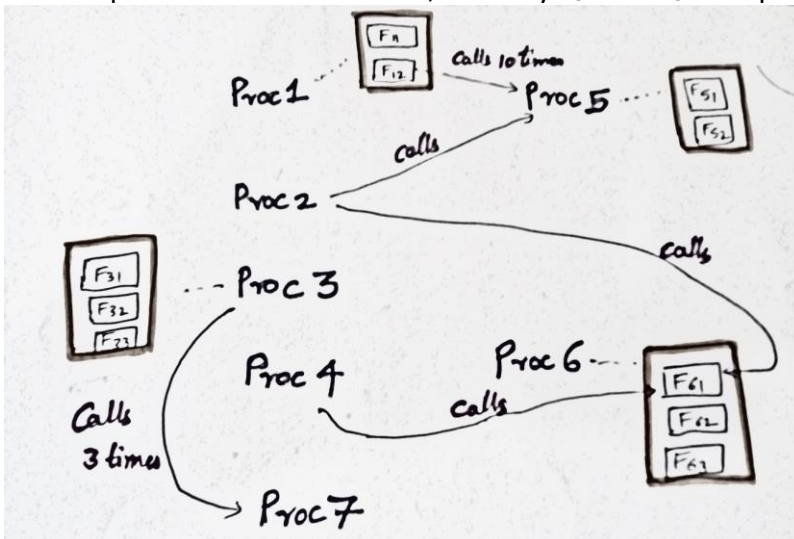
Q6. You are asked to design a system for a particular purpose and you find that there are already some solutions which exist in market which do that functionality. The details of those solutions in terms of their cost and computational power is given below (note that cost is the value specified in K Rs. Where K stands for thousand, whereas the computational capacity is in terms of GB RAM). Identify the pareto-optimal solutions among them by encircling them in table below. [2M]

Device	D1	D2	D3	D4	D5	D6	D7	D8
Cost (K Rs.)	90	80	30	60	30	20	80	60
Computational Power (RAM in GB)	6	3	11	9	5	6	9	5

Q7. Consider 5 nodes/subsystems with distance matrix given below. Apply hierarchal clustering using complete linkage based algorithm. After the evaluation specifically highlight the final dendrogram and the nested cluster diagram. [8M]

	P1	P2	P3	P4	P5
P1	0	0.7	5.6	3.6	3.2
P2	0.7	0	4.9	2.9	2.5
P3	5.6	4.9	0	2.2	2.4
P4	3.6	2.9	2.2	0	0.5
P5	3.2	2.5	2.4	0.5	0

Q8. Consider the following process having different Procedures as shown below. Apply a. procedure inlining, b. procedure exlining and c. procedure cloning to the most suitable places within the process. Mark the answers in the figure itself clearly highlighting them and justify your answers in the space below. Note that F₁₁ and F₂₂ are sub-procedures within Proc1, similarly F₅₁ and F₅₂ sub-procedures within Proc5 etc. [6M]



Q9. What are the four performance estimation methods? Please write them below, with 1-line description of each [4M]