National University of Computer and Emerging Sciences
Lahore Campus

Quiz 2 Solution

Q1: Read the information in the following table carefully.

Run-times of *Parallel Odd-Even sort* on a target machine / cluster (time is in milliseconds)		
No of Processors (Column below)	Number of integer keys to sort (in thousands)	
	200 K keys	**3200 K keys**
1	88 (milliseconds)	1800 (milliseconds)
2	43 (milliseconds)	860 (milliseconds)
4	22 (milliseconds)	430 (milliseconds)
8	12 (milliseconds)	220 (milliseconds)
16	7.5 (milliseconds)	130 (milliseconds)

Q1 a: Calculate the speedup of parallel sorting for 200K and 3200K keys.

Run-times of *Parallel Odd-Even sort* on a target machine / cluster (time is in milliseconds)		
No of Processors (Column below)	Number of integer keys to sort (in thousands)	
	Speedup for 200 K keys	**Speedup for 3200 K keys**
1	-	-
2	88/43=2.04	1800/860=2.09
4	88/22=4	1800/430=4.18
8	88/12=7.33	1800/220=8.18
16	88/7.5=11.73	1800/130=13.84

Q1 b: Comparing speed-ups for 200K and 3200K what can you infer?

(1) Speedups initially are good but then start flattening out (for example from 8 to 16 processors, ideally speedup should have doubled but it didn't.)

(2) We get better speedup with more data but same number of processors. So having more data, opportunity of parallelization might increase. (Gustafson's Law)

Q1 c: Consider a new metric defined as follows:

$$m = \frac{Speedup}{No.of\ processors\ used}$$

Calculate the metric m for both 200k and 3200K datasets.

Run-times of *Parallel Odd-Even sort* on a target machine / cluster (time is in milliseconds)		
No of Processors (Column below)	Number of integer keys to sort (in thousands)	
	m for 200 K keys	**Speedup for 3200 K keys**
1	-	-

2	88/43=2.04, m = 2.04/2=1.02	1800/860=2.09, m = 2.09/2=1.04
4	88/22=4, m = 4/4=1	1800/430=4.18, m = 4.18/4=1.05
8	88/12=7.33, m = 7.33/8=0.92	1800/220=8.18, m = 8.18/8=1.03
16	88/7.5=11.73, m = 11.73/16=0.73	1800/130=13.84, m = 13.84/16=0.87

Q1 d: Compare metric ***m*** for 200K and 3200K datasets? What can you infer?

(1) Within the same dataset, values of m are decreasing as we increased number of processors.

(2) Across datasets, and for the same number of processors, value of m is larger for larger dataset.

(3) Here “m” is efficiency of our parallelization. Its value close to 1 is good, decreasing value of m means that our parallelization is running out of stream and we are getting farther from the ideal speed-up.