David Perry CS 525 HW # 6 4/24/2014

1.)
a.)
10^6 Numbers

Number of Processors	Time	Speedup
1	0.805095911026001	N/A
2	0.421966075897217	1.907963594737192
4	0.228261947631836	3.527070190098182
8	0.177057981491089	4.547074942603025

10^7 Numbers

Number of Processors	Time	Speedup
1	9.343683004379272	N/A
2	4.829205989837646	1.9348280077598017
4	2.565811872482300	3.6416087650805467
8	1.556595087051392	6.002641972922265

b.)

4 Processors

N size	Time	Speedup	Efficiency
10^6	0.228360891342163	N/A	N/A
(10^6) * 2	0.498054027557373	N/A	N/A
(10^6) * 4	1.136424064636230	N/A	N/A
(10^6) * 8	2.028274059295654	N/A	N/A

8 Processors

N size	Time	Speedup	Efficiency
10^6	0.164587020874023	1.387478126339946	0.693739063169973
(10^6) * 2	0.348948955535889	1.4272976596032503	0.7136488298016251
(10^6) * 4	0.667689085006714	1.7020258233288368	0.8510129116644184
(10^6) * 8	1.272871971130371	1.5934627403999237	0.7967313701999619

Based on the results in experiment B it can be concluded that the sample sort algorithm has good isoefficiency. This conclusion is based on the fact that the observed per processor efficiency stays fairly constant as the problem size increase. This is expected as the sample sort algorithm divides the problem amongst processors. Therefore, assuming that the splitter selection is decent the workload should be evenly distributed.

2.)

a.)

$$T(n,1)=n\log n$$
$$T(n,P)=(\frac{n}{P})\log n+t_w(\frac{n}{P})\log P$$
$$T_{OH}(n,P)=P(T(n,P))-T(n,1)$$
$$n\log n\geq P(\frac{n}{P}\log n+t_w\frac{n}{P}\log P)-n\log n$$
$$\log n\geq C*t_w\log P$$

b.)

$$T(n,1)=n^2$$
$$T(n,P)=\frac{n^2}{P}+4(t_s+t_w\frac{n}{\sqrt{p}})$$
$$T_{OH}(n,P)=P(T(n,P))-T(n,1)$$
$$n^2\geq P(\frac{n^2}{P}+4\mathrm{t}_s+t_w\frac{4\mathrm{n}}{\sqrt{p}})-n^2$$
$$n^2\geq 4\mathrm{Pt}_s+4\mathrm{nt}_w\sqrt{p}$$

$$n^2\geq C*4\mathrm{Pt}_s$$
$$n\geq\sqrt{C*4\mathrm{Pt}_s}$$

$$n^2\geq C*(4\mathrm{nt}_w\sqrt{p})$$
$$n\geq\sqrt{C*(4\mathrm{nt}_w\sqrt{p})}$$

3.)

Complexity of the Parallel Algorithm:

Broadcast:

$$\sum_{i=1}^{logP} \log_2(P)(t_s+t_w)$$

$$\log_2^2(P)(t_s+t_w)$$

Partitioning:

$$\sum_{i=1}^{logP} (\frac{n}{p})$$

$$\log_2(P)(\frac{n}{P})$$

Transmission of partitions (Assuming that processes send and receive in one transmission):

$$\sum_{i=1}^{logP} (t_s+t_w \frac{n}{2P})$$

$$\log_2(P)*((t_s+t_w \frac{n}{2P}))$$

Sequential Quicksort:

$$(\frac{n}{p})\log_2(\frac{n}{p})$$

Total Runtime:

$$\log_2^2(P)(t_s+t_w) \quad + \quad \log_2(P)(\frac{n}{P}) \quad + \quad \log_2(P)*((t_s+t_w \frac{n}{2P})) \quad + \quad (\frac{n}{p})\log_2(\frac{n}{p})$$

Speed up:

$$\frac{n \log_2 n}{\log_2^2(P)(t_s+t_w)+\log_2(P)(\frac{n}{P})+\log_2(P)*((t_s+t_w \frac{n}{2P}))+(\frac{n}{p})\log_2(\frac{n}{p})}$$

Isoefficiency:

$$T(n,1)=n\log n$$
$$T(n,P)=\log_2^2(P)(t_s+t_w)+\log_2(P)(\frac{n}{P})+\log_2(P)*((t_s+t_w\frac{n}{2P}))+(\frac{n}{p})\log_2(\frac{n}{p})$$
$$T_{OH}(n,P)=P(T(n,P))-T(n,1)$$
$$T_{OH}(n,P)=(P*\log_2^2(P)(t_s+t_w)+\log_2(P)(n)+\log_2(P)*((Pt_s+t_w\frac{n}{2}))+(n)\log_2(\frac{n}{p}))-n\log_2 n$$
$$T_{OH}(n,P)=(P*\log_2^2(P)(t_s+t_w)+\log_2(P)(n)+\log_2(P)*((Pt_s+t_w\frac{n}{2}))+(n)(\log_2(n)-\log_2(p)))-n\log_2 n$$
$$T_{OH}(n,P)=(P*\log_2^2(P)(t_s+t_w))+\log_2(P)*((Pt_s+t_w\frac{n}{2}))$$

Therefore, there are 3 relationships to consider:

1.)
$$n\log n\geq P*\log_2^2(p)(t_s+t_w)$$

2.)
$$n\log n\geq\log_2(P)*Pt_s$$

3.)
$$n\log n\geq\log_2(P)*t_w\frac{n}{2}$$
$$\log n\geq\frac{\log_2(P)*t_w}{2}$$