

# Algorithms for Big Data

Spring Semester 2022

## Exercise Set 14

Here is a formalization of MPC model (one of many possible, equivalent):

- Input size  $N$ , distributed among machines.
- Machine memory is  $S = N^\alpha$  for some  $0 < \alpha < 1$ .
- Machines are numbered with unique ID's,  $1 \dots \frac{N}{S}$ .
- After each round machines send messages addressed to other machines. Each machine can send  $\mathcal{O}(S)$  atomic messages in total and receive  $\mathcal{O}(S)$  atomic messages in total.

### Exercise 1:

Show an algorithm for maximum computation: input array  $x[1 \dots N]$ . Output:  $\max\{x[i]\}$  (on a single machine), in time  $\mathcal{O}(\frac{1}{\alpha})$ .

### Exercise 2:

Show an algorithm for broadcasting: as an input one machine has a message  $m$  of size  $\mathcal{O}(S)$ . Output: all machines have  $m$ . Show  $\mathcal{O}(\frac{1}{\alpha})$  algorithm.

### Exercise 3:

Show that broadcasting cannot be done faster: that there is no  $o(\frac{1}{\alpha})$  algorithm.

### Exercise 4:

Show an algorithm for prefix sums: input array  $x[1 \dots N]$ . Output: array  $y[1 \dots N]$  where  $y[i] = x[1] + \dots + x[i]$ . Time:  $\mathcal{O}(\frac{1}{\alpha})$ .

### Exercise 5:

Show an algorithm for offsets: input array  $x[1 \dots N]$  and  $S$  values  $a_1, \dots, a_S$ . Output: values  $j_1, \dots, j_S$  where  $j_k$  is the position of  $a_k$  in sorted  $x[1 \dots N]$ . Time:  $\mathcal{O}(\frac{1}{\alpha})$ .

### Exercise 6:

Show an algorithm for pivot: input array  $x[1 \dots N]$  and  $S$  values  $a_1, \dots, a_{S-1}$ . Output: reshuffle  $x$  so that some prefix of machines holds all the values from  $x$  smaller than  $a_1$ , then next batch of machines holds all values from  $x$  between  $a_1$  and  $a_2$ , etc. Time:  $\mathcal{O}(\frac{1}{\alpha})$ .

### Exercise 7:

(2 pts.)

Sorting: input array  $x[1 \dots N]$ . Output:  $x$  sorted. Time:  $\mathcal{O}(\frac{1}{\alpha^2})$ . Idea:

- Pick sample of size  $S$ .
- Use it as a pivot.
- Show that whp subproblems are of size  $\tilde{\mathcal{O}}(\frac{N}{\sqrt{S}})$ .
- Recurse on subproblems.