

Partial Sums Problem

Given  $a_0, a_1, a_2, \dots, a_{n-1}$ ,

compute  $\sum_{i=0}^j a_i$  for  $j = 0, 1, 2, \dots, n-1$   
( $n$  results)

How to parallelize?

As with the primes generation, we could try several ideas; here we just show agenda - ~~parallel~~ parallelism.

Suppose  $k$  threads:

thread 1: process  $j = 0, 1, 2, \dots, n/k$

thread 2: process  $j = n/k + 1, \dots, 2n/k$

⋮  
etc.

Example: 3 threads

thread 1:

~~for i in range(0, n):~~

$r[0] = a[0]$

for  $i$  in range(1,  $n/3$ ):

$r[i] = a[i] + r[i-1]$

for  $i$  in range( $n/3$ ):

~~compute~~

Out(("psum",  $r[i]$ ,  $i$ ))

thread 2:

$$r[n/3] = a[n/3]$$

for  $i$  in range( $n/3+1, 2n/3$ ):

$$r[i] = a[i] + r[i-1]$$

$s = \text{In}(\text{"psum"}, \text{None}, n/3 - 2)$

for  $i$  in range( $n/3, 2n/3$ ):

$$\text{Out}(\text{"psum"}, r[i] + s[1], i)$$

thread 3:

$$r[2n/3] = a[2n/3]$$

for  $i$  in range( $2n/3+1, n$ ):

$$r[i] = a[i] + r[i-1]$$

$s = \text{In}(\text{"psum"}, \text{None}, 2n/3 - 1)$

for  $i$  in range( $2n/3, n$ ):

$$\text{Out}(\text{"psum"}, r[i], i) + s[1]$$

main:

for  $k$  in range( $n$ ):

$s = \text{In}(\text{"psum"}, \text{None}, k)$

print  $s[1]$

## Review of Topics

TCP/IP logical hierarchy

Ethernet, ARP, IP, TCP, UDP

Addresses & Names

MAC, IP, Port, FQDN

multiplexing, demultiplexing, relation between addresses.

Address Translation & Forwarding

wildcard matching, LAN forwarding

Headers, fields in packets

Sockets, clients and servers

Concurrency patterns for servers

The stream model of TCP

Concurrency control mechanisms

The HTTP protocol

Strategies for Parallel Programming

result-oriented, agenda, specialist styles.