

Haskell Functions		MPI	
map	map (+1) [1..5]	MPI_comm_size(- wrld, &out)	get size
filter	filter even [1..5]	MPI_comm_rank(wrld, &out)	get rank
take	take 2 [-2..10]	MPI_Scatter(data, amount, type, &local, , amount,..)	
drop	drop 1 [0,1..5]	MPI_reduce	a0=a0+a1+a2
head	head [1..5] = 1	MPI_allreduce	a0=sum(); a1=sum-()...
tail	tail [1..5] = [4..5]	MPI_scatterV(data, [amounts],...)	Variable amounts
all	all even [2,4,6]	MPI_gatherV(data, [amounts],...)	Variable amounts
any		MPI_ReduceScatter	Reduce to root then scatter to all
iterate		MPI_Alltoall	Transpose Data matrix
repeat		MPI_allGather	a=b=c
zip	zip [1..5] [5..9]		
elem	elem 4 [1..5]		
[(a,b) a <- [1..5], b <- [5..9]]			
(+1) 5	=6		
(\x ->4) 6	=6		
(\x y ->x+y)	=5		

Church Encoding	
CTrue	\t. \f. t
CFalse	\t. \f. f
C2	\s. \x. s (s x)
CNot	\a. a CFalse CTrue
CAnd	\a. \b. a a b
COr	\a. \b. a b a
CEven	\n. n not CTrue



By **tankman175**

cheatography.com/tankman175/

Not published yet.
 Last updated 7th April, 2022.
 Page 1 of 1.

Sponsored by **Readable.com**
 Measure your website readability!
<https://readable.com>