Verteilte Systeme, WiSe 2014 31.10.14
Tim-Ole Schaffeld, Hector Smith

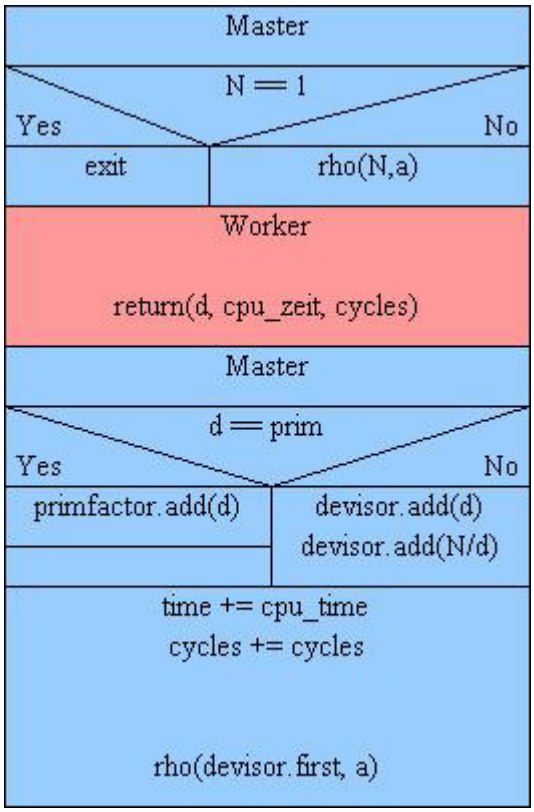
Labor 2

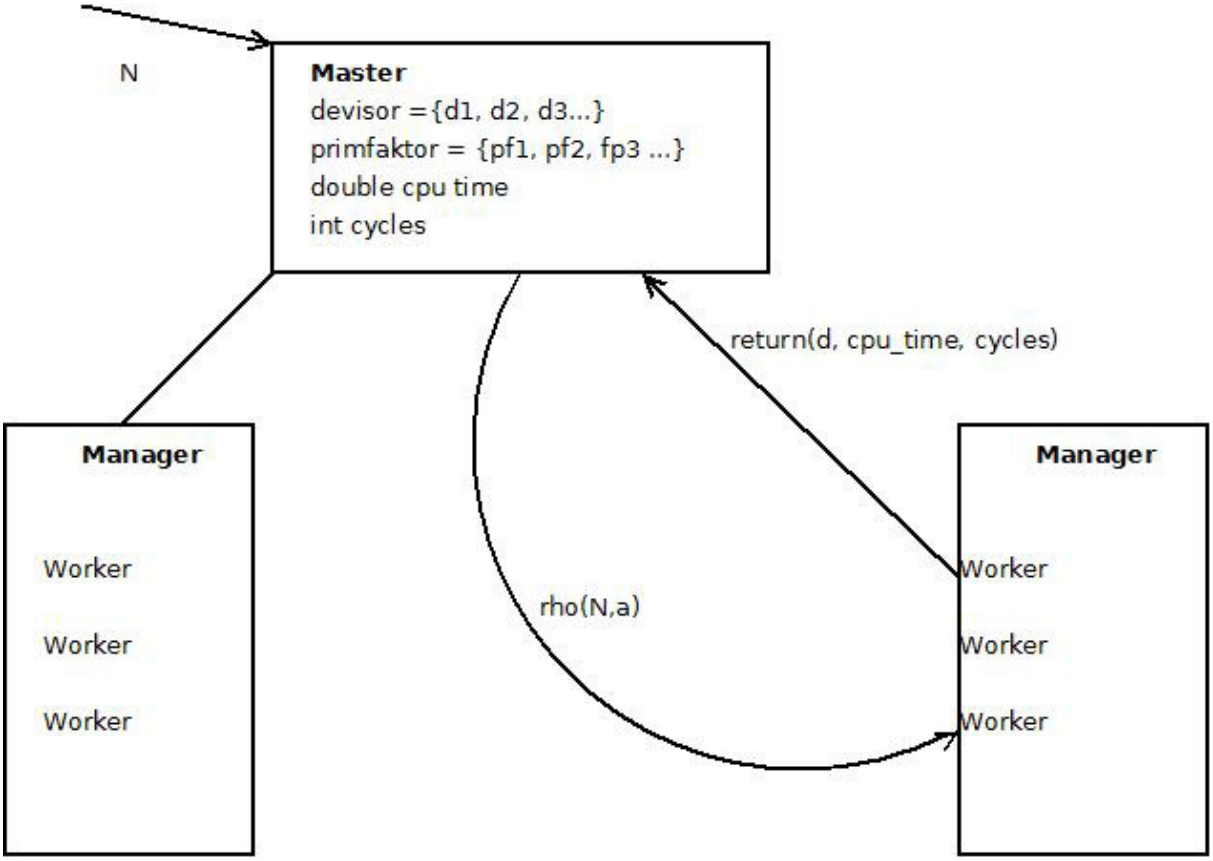
Konzeption

Ablauf:
Create Master
Master creates Manager
Manager creates Workers
Worker connects to Master
Master gets new N
Master creates pseudo random numbers
Master sends msg to each worker: rho(N, randomNumber)
First Worker returns devisor, cpu_time, cycles
Master checks if its prim:

Yes: Add primfaktor to primfactor list.
No: Store it in devisor list (queue) to rework on it
Send new task to all worker with next d and a new randomNumber

When devisor list is empty Master kills the Manager and prints out the primfactor list.





- Every Manager / Worker is connected to the Master.

Messages

NEWNUMBER , N	Master -> Worker
RETURNDEVISOR, d, cpu_time, cycles	Worker -> Master
SHUTDOWN	Master -> Manager

If the deadline is reached d= -1 is returned.

In our plan all the workers are in a race to find the next devisor.
The first worker who found one automaticly starts a new race for all workers with a new number.

Another possibility is only to send the founder a new task with a new number but then they are not "helping" each other and they may run up to their dealine.

To Do
Server unreachable
2 Worker finishing at the same time
Measure cputime vs wall clock time