Dept. Math. & Comp. Sc.	**Distributed Systems**
Vrije Universiteit	**09.08.1999**

1a Explain what is meant by *middleware*. *5pt*

1b Why is it normally so hard to scale traditional client/server applications? *5pt*

1c Under what conditions would you expect that server-initiated client-side caching in the Web would work? *5pt*

1d Some distributed file systems give clients a lease on cached entries. Why are leases sometimes useful? *5pt*

2a What is the difference between a hard link and a soft link in a naming system? *5pt*

2b Having only pure names as object references (i.e., names containing no information at all) how can you efficiently locate an object in a worldwide distributed systems? *10pt*

2c Explain what a closure mechanism is, and why such mechanisms are inherently implicit. *5pt*

3a Give a precise description of reliable multicasting. *5pt*

3b Why is reliable multicasting so hard to scale to large numbers of receivers? *10pt*

3c In virtual synchrony, a process $P[i]$ piggybacks $N[i][j]$ on its multicast message to the others, where $N[i][j] = N$ means that $P[i]$ has successfully received messages #1,...#N from process $P[j]$. A process $P[i]$ will *deliver* message #N from $P[j]$ only if it knows that all others have received #N as well. What global message ordering does this scheme impose. Explain your answer! *10pt*

4a Under which failure semantics do you need $2k+1$ members to implement a $k-$fault tolerant service? *5pt*

4b Give an example in which masking an omission failure leads to a performance failure. *5pt*

4c Explain how quorum-based replication works. *10pt*

4d What is the relation between quorum-based replication and group failure masking? *5pt*

Grading: *The final grade is calculated by accumulating the scores per question (maximum: 90 points), and adding 10 bonus points. The maximum total is therefore 100 points.*