1: True/False (1%)

Name:

Sett sirkel rundt the riktige svaralternativet.
Circle the correct answer.

1a

Bruk av prioritetsarv gjør vranglås umulig.
The use of priority inheritance excludes the possibility of deadlocks

A: True **B:** False

1b

Gitt at Rate Monotonic algoritmen for tildeling av prioriteter er benyttet: Dersom CPU-utnyttelsen er over 69,3% vil man med mange oppgaver alltid miste tidsfrister.
With the use of the rate-monotonic scheduling algorithm: If the CPU-utilization is above 69,3%, some deadlines will always be missed.

A: True **B:** False

1c

Når den opprinnelige prioritets-tak algoritmen benyttes, må man på forhånd kjenne oppgaven med høyest prioritet som benytter den aktuelle ressursen.
When the original priority ceiling protocol is used, one must, in advance, know the task with the highest priority that uses the actual resource.

A: True **B:** False

1d

EDF er en dynamisk prioritet scheduling algoritme.
EDF is a dynamic priority scheduling algorithm

A: True **B:** False

2: Scheduling Policies (1%)

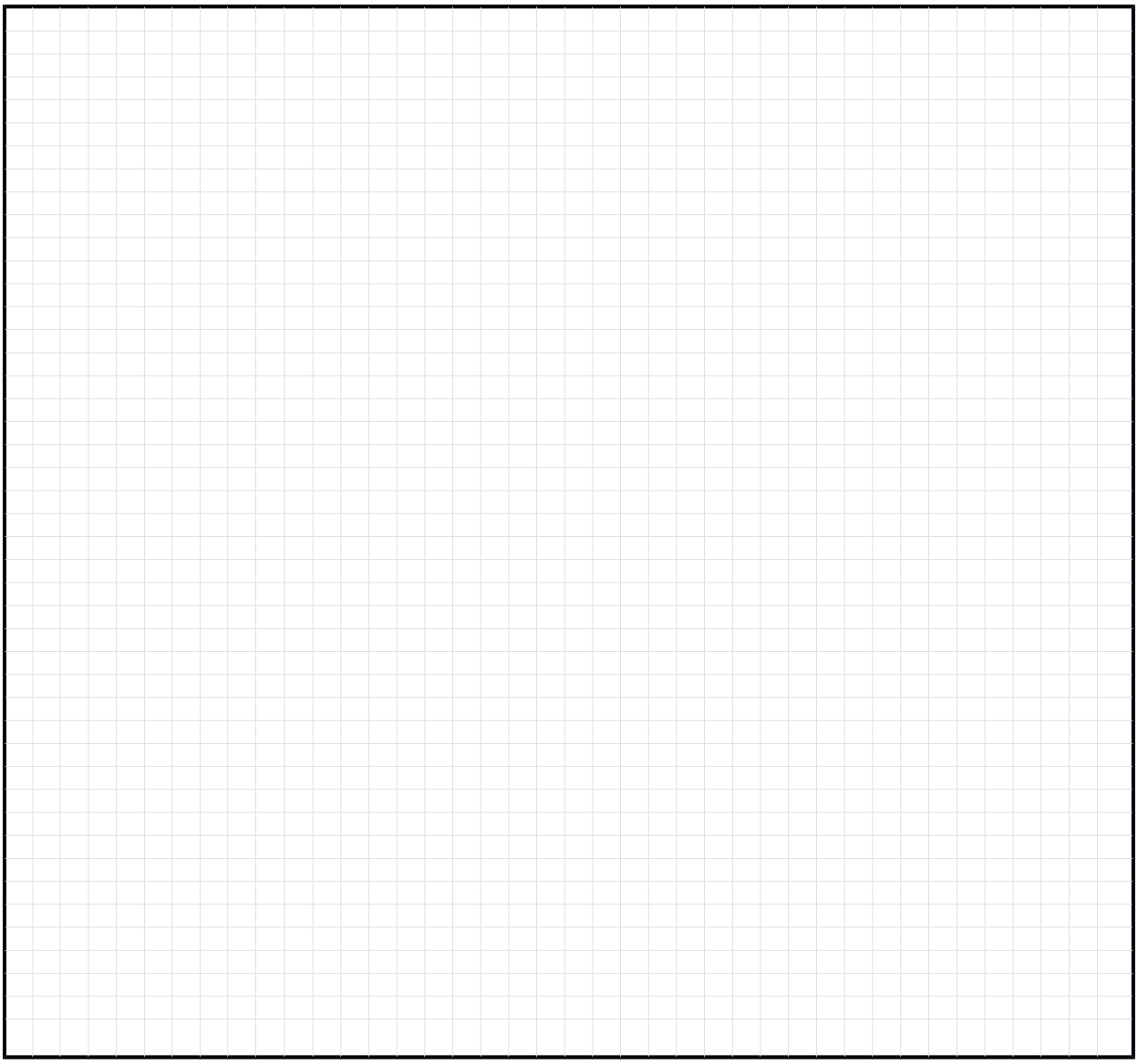
2a

Vis grafisk hvordan task a, b, c og d i Table vil bli utført dersom det brukes:
Show in a graphical way how the tasks a, b, c and d in Table will run with:

1. FIFO
2. Round-Robin (q=1)

Table 0

Tasks	Arrival time	Exec.time
A	0	5
B	2	2
C	4	4
D	6	3

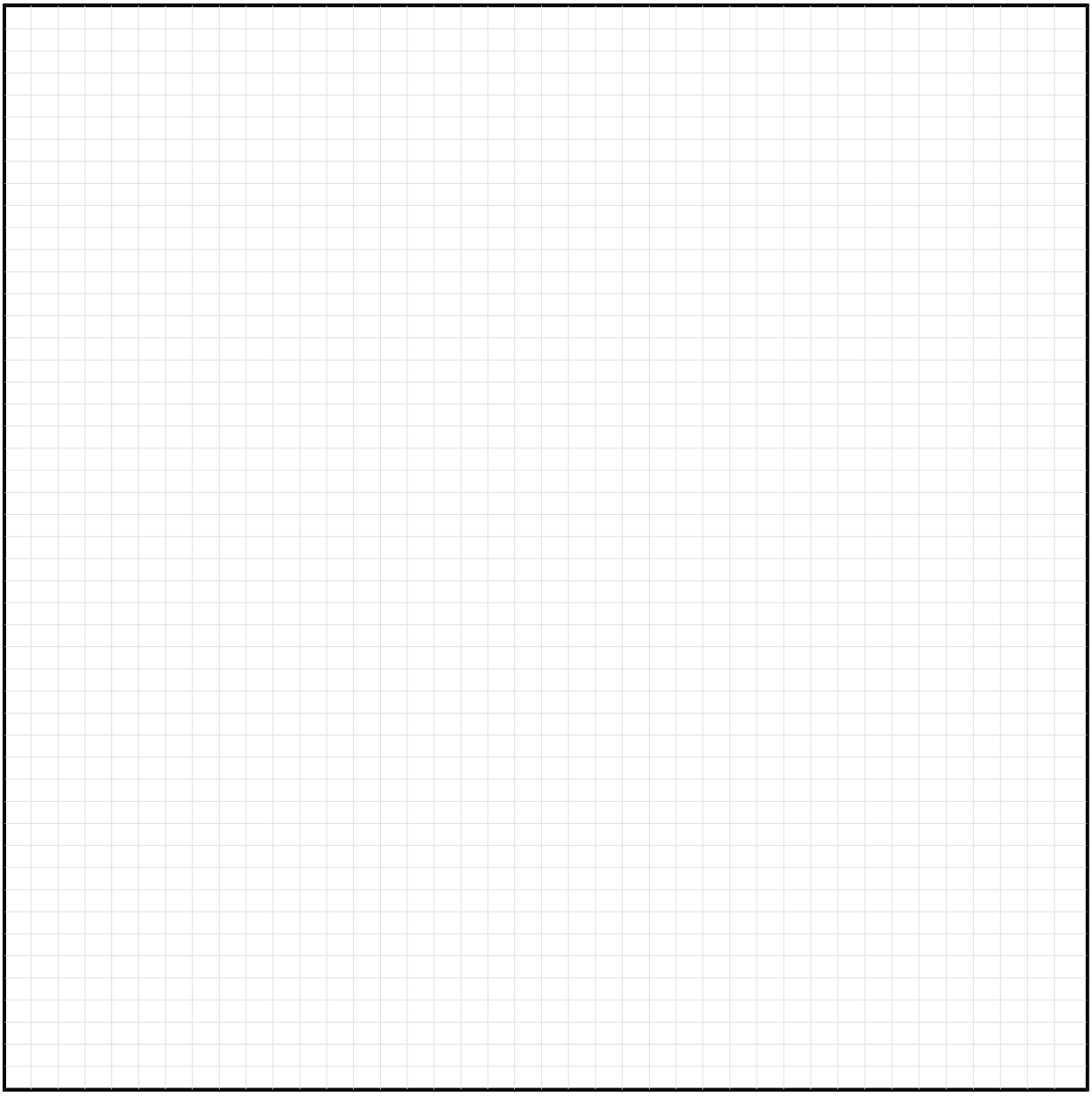


2b

Vis grafisk hvordan task a, b, c og d i Table vil bli utført dersom det brukes preemtive fixed-priority scheduling.
Show in a graphical way how the tasks a, b, c and d in Table will run with preemtive fixed-priority scheduling.

Table 0

Tasks	Arrival time	Exec.time	Priorities
A	0	5	1
B	2	2	2
C	4	4	4
D	6	3	3



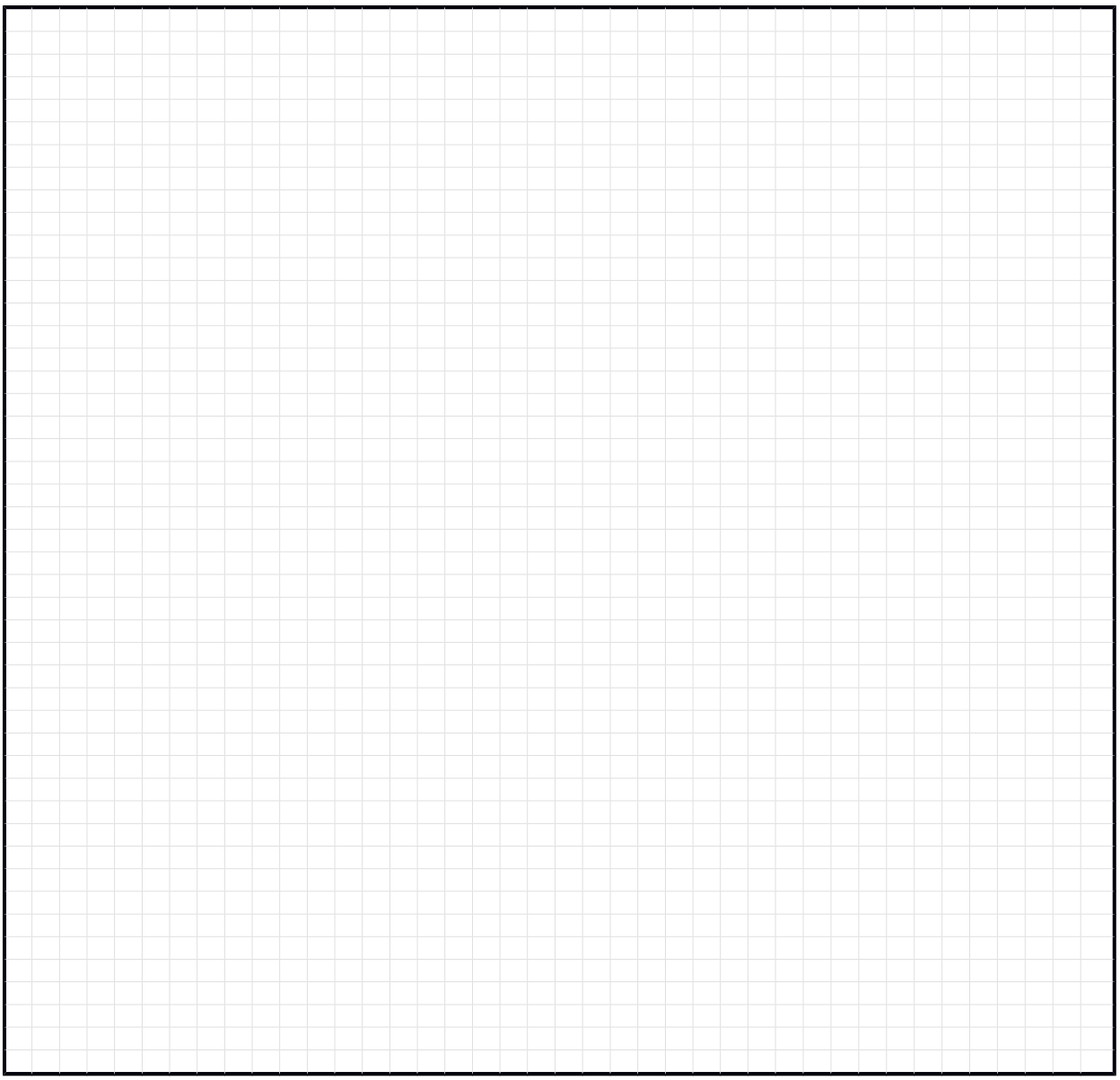
3: Priority Inversion (1%)

3a

Vis grafisk hvordan task a, b, c og d i Table 0 vil bli utført dersom de følger den oppgitte sekvensen, når det ikke benyttes prioritetsarv.
Show in a graphical way how the tasks a, b, c and d in Table 0 will run when following the execution sequence, without priority inheritance.

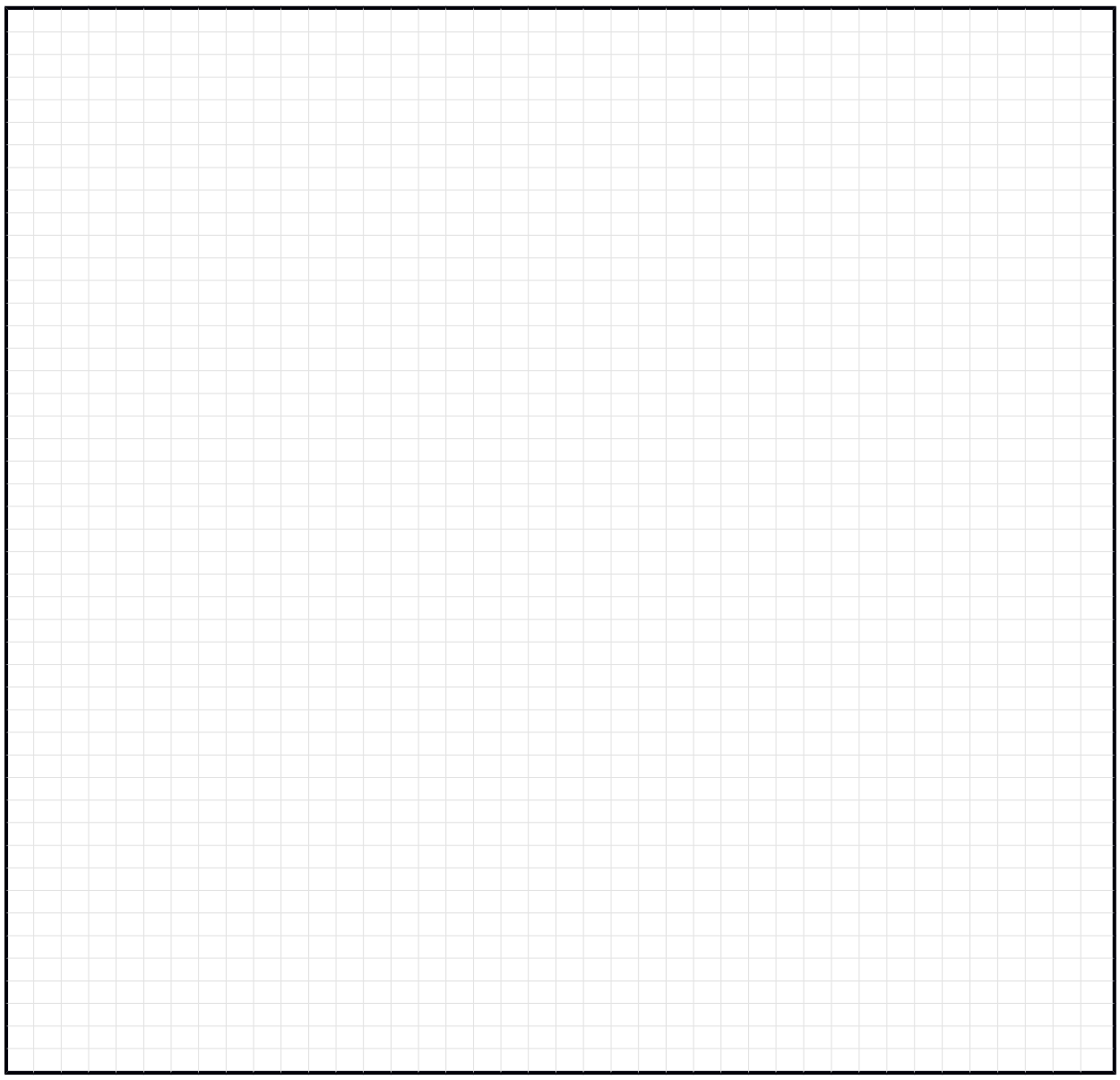
Table 0 (E=Executing, Q=Executing with resource "Q" locked)

Tasks	Arrival time	Exec.time	Priorities	Sequence
A	0	5	1	EQQQE
B	2	2	2	EE
C	4	4	4	EEQQ
D	6	3	3	EEE



3b

Gjør det samme som i oppgave 3a, men med prioritetsarv.
Do the same as in exercise 3a, but with priority inheritance.



4: Utililization and Response Time (2%)

Taskene A, B og C i Table har fordelt prioriteter med rate monotonic. Kan du si om de vil klare å møte sine tidsfrister ved å bruke:
The tasks A, B and C in Table have assigned priorities with rate monotonic. Can you say whether the tasks will meet their deadlines by using:

1. Utilization-based schedulability test
2. Response time analysis

Table 0

Task	Period	Exec.time
A	50	12
B	20	6
C	40	10

Formulas:

$$\sum_{i=1}^{N}\left(\frac{C_i}{T_i}\right)\leq N\left(2^{\frac{1}{N}}-1\right)$$

$$R_i=C_i+\sum_{j\in hp(i)}\lceil\frac{R_i}{T_j}\rceil C_j$$

N	N(2^(1/N) – 1)
1	1
2	0.828
3	0.780
4	0.757