```
# COMP9315 21T1 Final Exam Q6

Type your answer(s) to replace the xxx's
Submit this file as your answer for Q6

a.

Minimum cost occurs if we keep R as the outer relation.
In this case, all bS pages of table S can go into the buffers,
because 25 -2 >= 20 (N - 2 >= bS).
So, we can hold the entire S relation in memory and join it
with bR pages of R one by one.
Total minimum cost = bR + bS = 20 + 40 = 60

The minimum number of buffers would be the buffers needed to
hold the entire S relation (20),
plus one buffer for reading table R,
and one buffer for writing the output.

Minimum number of buffers required = bS + 2 = 20 + 2 = 22.


b.
Cost for sorting R = 2.bR (1 + ceil(logN-1(bR /N)))
                   = 2 * 40 * (1 + ceil(log24(40/25)))
                   = 2 * 40 * (1 + ceil(log24(2)))
                   = 2 * 40 * (1 + 1)
                   = 2 * 40 * 2
                   = 160
Cost for sorting S = 2.bS (1 + ceil(logN-1(bS /N)))
                   = 2 * 20 (1 + ceil(log24(20/25)))
                   = 2 * 20 (1 + ceil(log24(1)))
                   = 2 * 20 ( 1+ 0)
                   = 2 * 20
                   = 40

Cost for merging R and S = bR + bS (best case scenario)
                         = 40 + 20
                         = 60


Total cost = 160 + 40 + 60 = 260

Minimum buffers needed to achieve this cost is 20.
If we have less than 20 buffers, we would need to
do another pass to sort S.

Consider e.g. N = 19,
ceil(logN-1(bS/N)) = ceil(log18(20/19)) = ceil(log18(2)) = 1
But for N = 20,
ceil(logN-1(bS/N)) = ceil(log19(20/20)) = ceil(log19(1)) = 0
```

```
So, at N = 20, the sorting cost is minimised, and any number
of buffers greater than 20 will have the same cost as N = 20.


c.

Cost for sorting S = 40
Cost for merging R and S = 60
Total cost = 40 + 60 = 100

Minimum buffers needed to achieve this cost would be 20.
Same reasoning as for (b).


d.

Cost for sorting R = 160
Cost for merging R and S = 60
Total cost = 160 + 60 = 220

In this case, since S is already sorted,
the minimum number of buffers needed is dependent on R.
The minimum number is 7.

At N = 7, ceil(logN-1(bR /N)) = ceil(log6(40/7)) = ceil(log6(6)) = 1
At N = 6, ceil(logN-1(bR /N)) = ceil(log5(40/6)) = ceil(log5(7)) = 2

e.

Total cost is just the cost of merging them = bR + bS = 40 + 20 = 60
In this case, the minimum buffers needed would be just 3. One to hold
each relation, and one for output.
This is assuming that there are no 'runs' in the inner relation that
span more than one page.
```