

(a) List all of the input variables, including the (abstract) state variables.

| | |
|-----------------|----------------------|
| input variables | capacity, element |
| state variables | size, front, back |

(b) Define the characteristics of the input variables. Make sure you cover all input variables.

| Method | Params | Returns | Values | Exception | Ch ID | Characteristic | Covered by |
|--------------------|----------|---------|-------------|-------------------------|-------|-------------------------------------|------------|
| constructor | capacity | | | | C1 | Constructor | |
| | | | | capacity is less than 0 | C2 | If capacity is larger or equal to 0 | |
| enqueue | element | | | | C3 | Push element into the back of queue | |
| | | | | element is invalid | C4 | Element is valid | |
| | | | | queue is full | | | C7 |
| dequeue | state | object | object o | | C5 | Get the first element from queue | |
| | | | | queue is empty | | | C6 |
| is_empty | state | boolean | True, False | | C6 | The queue is non-empty | |
| is_full | state | boolean | True, False | | C7 | The queue is not full | |

(c) Partition the characteristics into blocks. Designate one block in each partition as the "Base" block.

| ID | Characteristic | constructor(capacity) | enqueue(element) | dequeue() | is_empty() | is_full() |
|----|-------------------------------------|-----------------------|------------------|-----------|------------|-----------|
| C1 | Constructor | x | | | | |
| C2 | If capacity is larger or equal to 0 | x | | | | |
| C3 | Push element into the back | | x | | | |

| | | | | | | |
|----|----------------------------------|------|-------|------|-----|-----|
| | of queue | | | | | |
| C4 | Element is valid | | x | | | |
| C5 | Get the first element from queue | | | x | | |
| C6 | The queue is non-empty | | | x | x | |
| C7 | The queue is not full | | x | | | x |
| | Base block | {TT} | {TTT} | {TT} | {T} | {T} |

(d) Define values for each block.

| | |
|----|---------------|
| C1 | {true, false} |
| C2 | {true, false} |
| C3 | {true, false} |
| C4 | {true, false} |
| C5 | {true, false} |
| C6 | {true, false} |
| C7 | {true, false} |

(e) Define a test set that satisfies Base Choice Coverage (BCC). Write your tests with the values from the previous step. Be sure to include the test oracles.

| Method | Characteristic | Test requirements | Infeasible TRs | Revised TRs | # TRs |
|-------------|----------------|----------------------|----------------|--|-------|
| constructor | C1, C2 | {TT, TF, FT} | TF, FT | TF -> FF FT -> FF | 2 |
| enqueue | C3, C4, C7 | {TTT, FTT, TFT, TTF} | FTT, TFT, TTF | TFT -> FFT TTF -> FTF FTT -> FFF | 4 |
| dequeue | C5, C6 | {TT, FT, TF} | FT, TF | FT -> FF TF -> FF | 2 |
| is_empty | C6 | {T, F} | None | | 2 |
| is_full | C7 | {T, F} | None | | 2 |