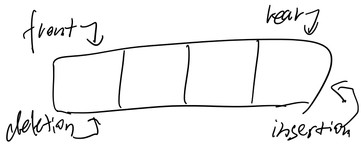
# Queue by Array

Sunday, 12 February 2023 23:03




#define SIZE 10

DATA_TYPE queue[SIZE];

functions:

① initialize

② insert

③ deleteElement

④ peek

⑤ display.

auxiliary functions:

⑥ reset

Explanations:

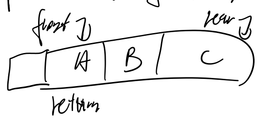
① initialize(DATA_TYPE* queue)

front = rear = -1 // index of elements.

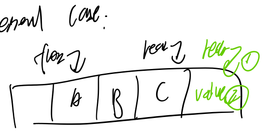
② insert(DATA_TYPE* queue, DATA_TYPE value)

special case: empty i.e. front == rear == -1


special case: full, i.e. rear == SIZE-1



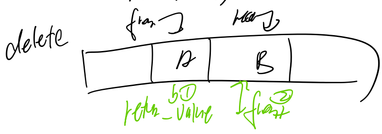

General case:




③ deleteElement(DATA_TYPE* queue)

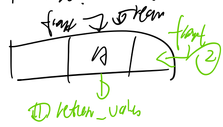
special case: empty i.e. front == rear == -1

return -1;




After deletion:

special case: only one element., i.e. front > rear; after deletion, it becomes empty queue.




then we need to reset the queue

reset(queue).

return return_value.

④ peek(DATA_TYPE* queue)

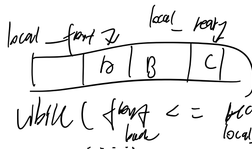
special case: empty. i.e. front == rear == -1.

return -1;

return queue[front].

⑤ display(DATA_TYPE* queue)

special case: empty. do nothing.




while(front <= rear) (local_front <= local_rear)

local_front++

⑥ reset(DATA_TYPE* queue)

initialize(queue).