```
class ArrayQueue:

    Default_Capacity = 10

    def __init__(self):
        self._data = [None]*ArrayQueue.Default_Capacity
        self._size = 0
        self._front = 0

    def __len__(self):
        return self._size

    def is_empty(self):
        return self._size == 0

    def first(self):
        if self.is_empty():
            raise Empty("Queue is Empty")
        return self._data[self._front]

    def enqueue(self,element):
        if self._size == len(self._data):
            self._resize(2*len(self._data))
        avail = (self._front + self._size)%len(self._data)
        self._data[avail]=element
        self._size += 1

    def _resize(self, limit):
        old = self._data
        self._data = [None]*limit
        start = self._front
        for i in range(self._size):
            self._data[i] = old[start]
            start = (1+start)%len(old)
        self._front = 0

    def dequeue(self):
        if self.is_empty():
            raise Empty("Queue is Empty")
        answer =self._data[self._front]
        self._data[self._front]=None
        self._front = (self._front+1)%len(self._data)
        self._size -= 1
        return answer
```