

실습) 다음의 양방향 원형 연결리스트로 큐를 구현한 C 코드를 Java 로 구현 하시오.

```
struct node {
    int item;
    struct node *llink;
    struct node *rlink;
};

void addq(struct node *, int);
void deleteq(struct node *);
void listprint(struct node *);

int main()
{
    struct node *head;
    int cond, item;

    head = (struct node *)malloc(sizeof(struct node)); //더미 노드
    head->llink = head->rlink = head; //초기화

    while(1)
    {
        printf("1:insert 2:delete 3:output 4:exit\n");
        scanf("%d", &cond);
        if (cond == 1)
        {
            printf("Input Item\n");
            scanf("%d", &item);
            addq(head, item);
        }
        else if (cond == 2)
            deleteq(head);
        else if (cond == 3)
            listprint(head);
        else if (cond == 4)
            break;
        else {
            printf("Wrong Input\n");
            exit(1);
        }
    }
    return 0;
}
```

```

void addq(struct node *head, int item)
{
    struct node *node, *newnode;
    newnode = (struct node *)malloc(sizeof(struct node));
    newnode->item = item;

    node = head->llink; // 삽입 위치
    newnode->llink = node;
    newnode->rlink = node->rlink;
    node->rlink->llink = newnode;
    node->rlink = newnode;
}

void deleteq(struct node *head)
{
    struct node *deleted;

    if ((head == head->llink) && (head == head->rlink)) // 공백큐 조건
    {
        printf("Queue Empty\n");
        exit (1);
    }
    else
    {
        deleted = head->rlink; //삭제 위치
        deleted->llink->rlink = deleted->rlink;
        deleted->rlink->llink = deleted->llink;
        free(deleted);
    }
}

void listprint(struct node *head)
{
    struct node *ptr;
    ptr = head->rlink;
    while (ptr != head)
    {
        printf("%d\n", ptr->item);
        ptr = ptr->rlink;
    }
}

```