

Double Circular Queue

จัดทำโดย

นาย ณภพ คุ่มขวานา รหัสนิสิต 6530200100

นางสาว ฮาบีบะห์ มะแซ รหัสนิสิต 6530200550

นางสาว ธวัลรัตน์ พิทักษ์ รหัสนิสิต 6530200657

นางสาว บุญพิทักษ์ ผมเพชร รหัสนิสิต 6530200681

นางสาว เพชรรัตน์ ทองล้วน รหัสนิสิต 6530200746

นางสาว อรณิชา ศรีสมาน รหัสนิสิต 6530200878

เสนอ

ผศ.ดร. จีรวรรณ เจริญสุข

คณะวิทยาศาสตร์ ศรีราชา

มหาวิทยาลัย เกษตรศาสตร์ วิทยาเขตศรีราชา

ไฟล์ทั้งหมดในการทำโปรแกรม

app2.c -> ใช้ในการ run app เป็นการเรียกใช้ 2 ไฟล์ข้างล่าง

DoubleCircularQueue.c -> เป็น function กระบวนการทั้งหมด ของ DCQ

System.c -> ใช้ในการแสดงผล และ ดัก input ต่างๆ

ช่องทางเพิ่มเติม หากต้องการนำไปใช้งาน

Github : [NutNaphop/Circular_Array_C: Using for Project \(github.com\)](https://github.com/NutNaphop/Circular_Array_C_Using_for_Project)

app2.c

```
1 #include <stdio.h>
2 #include <string.h>
3 #include <stdlib.h>
4 #include <ctype.h>
5 #include "DoubleCircularQueue.c"
6 #include "System.c"
7
8 int isRunning = 1;
9 int val;
10 char input[2];
11
12 int main() {
13     while (isRunning) {
14         clear() ;
15         displayMenu();
16         printf("Enter your choice: ");
17         scanf("%s", input);
18         while (getchar() != '\n') ; // Using for get input only one word , with out whitespace
19         clear() ;
20         if (isInt(input)) {
21             switch (input[0]) {
22                 case '1' :
23                     displayEF() ;
24                     printf("Enter the number you want to enqueue at the front: ");
25                     scanf("%s", input);
26                     if (isInt(input)) {
27                         val = atoi(input);
28                         enqueueFront(val);
29                     }
30                     else {
31                         printf("Invalid input. Please enter a valid integer.\n");
32                     }
33                     break;
34                 case '2':
35                     displayER() ;
36                     printf("Enter the number you want to enqueue at the rear: ");
37                     scanf("%s", input);
38                     if (isInt(input)){
39                         val = atoi(input);
40                         enqueueRear(val);
41                     }
42                     else {
43                         printf("Invalid input. Please enter a valid integer.\n");
44                     }
45                     break;
46                 case '3':
47                     displayDF() ;
48                     dequeueFront();
49                     break;
50
51                 case '4':
52                     displayDR() ;
53                     dequeueRear();
54                     break;
55
56                 case '5':
57                     displayItem() ;
58                     showItem() ;
59                     break;
60                 case '6':
61                     displayEXIT() ;
62                     isRunning = 0;
63                     printf("Exiting the program.\n");
64                     break;
65
66                 default:
67                     printf("Invalid choice. Please try again.\n");
68                     break;
69             }
70             wait() ;
71         }
72         else {
73             printf("Invalid input. Please enter a valid integer.\n");
74             wait() ;
75         }
76     }
77 }
```

DoubleCircularQueue.c

```
1 #include <stdio.h>
2 #include <stdlib.h>
3
4 #define SIZE 5
5 int items[SIZE];
6 int front = -1, rear = -1;
7
8 // Function
9 int enqueueFront(int data);
10 int enqueueRear(int data);
11 int dequeueFront();
12 int dequeueRear();
13 void showItem();
14 int isFull();
15 int isEmpty();
16
17
18 int enqueueFront(int data) {
19     if (isFull()) {
20         printf("Queue is full. Cannot enqueue at the front.\n");
21         return 0;
22     }
23     if (isEmpty()) {
24         front = 0;
25         rear = 0;
26     }
27     else {
28         front = (front - 1 + SIZE) % SIZE;
29     }
30     items[front] = data;
31     printf("Enqueued %d at the front. Enqueue Front Complete.\n", items[front]);
32     showItem();
33 }
34
35 int enqueueRear(int data) {
36     if (isFull()) {
37         printf("Queue is full. Cannot enqueue at the rear.\n");
38         return 0;
39     }
40     if (isEmpty()) {
41         front = 0;
42         rear = 0;
43     }
44     else {
45         rear = (rear + 1) % SIZE;
46     }
47     items[rear] = data;
48     printf("Enqueued %d at the rear. Enqueue Complete.\n", items[rear]);
49     showItem();
50 }
51
52 int dequeueFront() {
53     if (isEmpty()) {
54         printf("Queue is empty. Cannot dequeue from the front.\n");
55         return 0;
56     }
57     printf("Dequeued item: %d. Dequeue Complete.\n", items[front]);
58     if (front == rear) {
59         front = -1;
60         rear = -1;
61     }
62     else {
63         front = (front + 1) % SIZE;
64     }
65     showItem();
66 }
67
68 int dequeueRear() {
69     if (isEmpty()) {
70         printf("Queue is empty. Cannot dequeue from the rear.\n");
71         return 0;
72     }
73     printf("Dequeued item: %d. Dequeue Complete.\n", items[rear]);
74     if (front == rear) {
75         front = -1;
76         rear = -1;
77     }
78     else {
79         rear = (rear - 1 + SIZE) % SIZE;
80     }
81     showItem();
82 }
83
84 void showItem() {
85     printf("*****\n");
86     printf("Front -> %d\n", front);
87
88     if (isEmpty()) {
89         printf("Queue is empty. Nothing to display.\n");
90     }
91
92     else {
93         printf("\n");
94         for (int i = front; i != rear; i = (i + 1) % SIZE) {
95             printf("%d -> ", items[i]);
96         }
97         printf("%d", items[rear]);
98     }
99     printf("\n\nRear -> %d\n", rear);
100     printf("*****\n");
101 }
102
103 int isFull() {
104     if ((rear + 1) % SIZE == front) {
105         return 1;
106     }
107     return 0;
108 }
109
110 int isEmpty() {
111     if (front == -1 && rear == -1) {
112         return 1;
113     }
114     return 0;
115 }
```

System.c

```
1 #include <stdio.h>
2 #include <stdlib.h>
3 #include <ctype.h>
4
5 void clear();
6 void wait();
7 int isInt(char *str);
8 void clear();
9 void wait();
10
11 void displayMenu() {
12     printf("\n");
13     printf("***** Double Circular Queue *****\n");
14     printf("[1] Enqueue at Front\n");
15     printf("[2] Enqueue at Rear\n");
16     printf("[3] Dequeue from Front\n");
17     printf("[4] Dequeue from Rear\n");
18     printf("[5] Display Queue\n");
19     printf("[6] Exit\n");
20     printf("*****\n");
21 }
22
23 void displayEF() {
24     printf("\n");
25     printf("***** Double Circular Queue *****\n");
26     printf("[1] Enqueue at Front\n");
27     printf("\n");
28     printf("\n");
29     printf("\n");
30     printf("\n");
31     printf("\n");
32     printf("*****\n");
33 }
34
35 void displayER() {
36     printf("\n");
37     printf("***** Double Circular Queue *****\n");
38     printf("\n");
39     printf("[2] Enqueue at Rear\n");
40     printf("\n");
41     printf("\n");
42     printf("\n");
43     printf("\n");
44     printf("*****\n");
45 }
46
47 void displayOF() {
48     printf("\n");
49     printf("***** Double Circular Queue *****\n");
50     printf("\n");
51     printf("\n");
52     printf("[3] Dequeue from Front\n");
53     printf("\n");
54     printf("\n");
55     printf("\n");
56     printf("*****\n");
57 }
58
59 void displayOR() {
60     printf("\n");
61     printf("***** Double Circular Queue *****\n");
62     printf("\n");
63     printf("\n");
64     printf("\n");
65     printf("[4] Dequeue from Rear\n");
66     printf("\n");
67     printf("\n");
68     printf("*****\n");
69 }
70
71 void displayItem() {
72     printf("\n");
73     printf("***** Double Circular Queue *****\n");
74     printf("\n");
75     printf("\n");
76     printf("\n");
77     printf("\n");
78     printf("[5] Display Queue\n");
79     printf("\n");
80     printf("*****\n");
81 }
82
83 void displayEXIT() {
84     printf("\n");
85     printf("***** Double Circular Queue *****\n");
86     printf("\n");
87     printf("\n");
88     printf("\n");
89     printf("\n");
90     printf("\n");
91     printf("[6] Exit\n");
92     printf("*****\n");
93 }
94
95 int isInt(char *str) {
96     while (*str) {
97         if (!isdigit(*str)) {
98             return 0;
99         }
100         str++;
101     }
102     return 1;
103 }
104
105 void clear() {
106     system("cls");
107 }
108
109 void wait() {
110     system("pause");
111 }
112
```

Example of program display

#การแสดงผลทางหน้าจอเมื่อเปิดโปรแกรม

```
***** Double Circular Queue *****
* [1] Enqueue at Front *
* [2] Enqueue at Rear *
* [3] Dequeue from Front *
* [4] Dequeue from Rear *
* [5] Display Queue *
* [6] Exit *
*****
Enter your choice:
```

#เมื่อเลือก [1] Enqueue at Front

```
***** Double Circular Queue *****
* [1] Enqueue at front *
* * *
* * *
* * *
* * *
*****
Enter the number you want to enqueue at the front: 5
Enqueued 5 at the front. Enqueue Front Complete.
*****
Front -> 0
5
Rear -> 0
*****
Press any key to continue . . .
```

#เมื่อเลือก [2] Enqueue at Rear

```
***** Double Circular Queue *****
*
* [2] Enqueue at Rear
*
*
*
*
*
*****
Enter the number you want to enqueue at the rear: 7
Enqueued 7 at the rear. Enqueue Complete.
*****
Front -> 0

5 ->7

Rear -> 1
*****
Press any key to continue . . .
```

#เมื่อเลือก [3] Dequeue at Front

```
***** Double Circular Queue *****
*
*
* [3] Dequeue from Front
*
*
*
*
*****
Dequeued item: 1. Dequeue Complete!
*****
Front -> 0

5 ->7

Rear -> 1
*****
Press any key to continue . . .
```


#เมื่อเลือก [4] Dequeue at Rear

```
***** Double Circular Queue *****
*                                                                           *
*                                                                           *
*                                                                           *
* [4] Dequeue from Rear                                                  *
*                                                                           *
*                                                                           *
*****
Dequeued item: 7. Dequeue Complete!
*****
Front -> 0

5

Rear -> 0
*****
Press any key to continue . . .
```

#เมื่อเลือก [5] Display Queue

```
***** Double Circular Queue *****
*                                                                           *
*                                                                           *
*                                                                           *
*                                                                           *
* [5] Display Queue                                                      *
*                                                                           *
*****
Front -> 3

99 ->1 ->5 ->41 ->74

Rear -> 2
*****
Press any key to continue . . . █
```


#เมื่อ Input ค่าที่ไม่ใช่ Integer

```
***** Double Circular Queue *****
* [1] Enqueue at Front *
* [2] Enqueue at Rear *
* [3] Dequeue from Front *
* [4] Dequeue from Rear *
* [5] Display Queue *
* [6] Exit *
*****
Invalid input. Please enter a valid integer.
Press any key to continue . . .
```

```
***** Double Circular Queue *****
* [1] Enqueue at front *
* *
* *
* *
* *
* *
* *
*****
Enter the number you want to enqueue at the front: 1.1
Invalid input. Please enter a valid integer.
Press any key to continue . . .
```

```
***** Double Circular Queue *****
* *
* [2] Enqueue at Rear *
* *
* *
* *
* *
* *
*****
Enter the number you want to enqueue at the rear: 1af
Invalid input. Please enter a valid integer.
Press any key to continue . . .
```