

データ構造と基本アルゴリズム

report2

提出日：11月21日（月）

所属：工学部情報工学科

学籍番号：135713B

氏名：天願 寛之

演習問題 3.5 を参考にして、配列によってスタック、キューを実現せよ。ただし、演習問題の解答の通りではキューはうまく動作しないので注意すること。作成したプログラムの各行にコメントとして説明を加えること。また、プログラムが正しく機能していることを示す実行例を含めること。

作成したプログラム

1-a.stuck によるプログラム(MAX==5)

```
01 #include <stdio.h>
02 #define MAX 5 /*MAX の値を 5 と定義*/
03 int s[MAX],top;
04
05 main(){
06     int n,c;
07     initstack(); /*関数 initstack の呼び出し*/
08     printf("push: 1, pop: 2\n");
09     while(scanf("%d",&c)!= EOF){ /*while 文,入力された値が EOF ではない間ループする*/
10         switch(c){ /*switch-case 文*/
11             case 1: /*変数 c の値が 1 の場合 */
12                 printf("Input key\n"); /*追加する値の入力を促す */
13                 scanf("%d",&n); /*関数 scanf で入力された値を変数 n に代入し,*/
14                 push(n); /*変数 n を引数として関数 push を呼び出し, */
15                 break; /*一通り処理を終えると break */
16             case 2: /*変数 c の値が 2 の場合 */
17                 if(stack_is_empty()==1){ /*if 文,関数 stack_is_empty を呼び出し, */
18                     printf("Stack is empty!!\n"); /*返された値が 1 の場合関数 printf の出力,*/
19                     break; /*if 文の処理が終わると break */
20                 }
21                 printf("key: %d\n",pop()); /*返された値が 1 ではない場合, */
22                 break; /*関数 printf の出力.(関数 pop により返された値を*/
23             } /*整数として出力). 処理を終えると break */
24             printf("push: 1, pop: 2\n");
25         }
26     }
27
28     int stack_is_empty(){ /*関数 stack_is_empty */
29         if(top==0) return(1); /*if 文.top の値が 0 の場合,値 1 を返す*/
30         return(0); /*top の値が 0 ではない場合,値 0 を返す*/
31     }
32
33     initstack(){ /*関数 initstack */
34         top=0; /*top の値に 0 を代入*/
35     }
36
37     push(int n){ /*関数 push,引数 n */
38         s[top]=n; /*配列 s[top]に引数として受け取った変数 n を代入*/
39         top++; /*top をインクリメント */
40     }
41
42     int pop(){ /*関数 pop */
43         int key; /*top をデクリメント */
44         top--; /*変数 key に配列 s[top]を代入*/
45         key=s[top]; /*変数 key を返す */
46         return(key);
47     }
```

1-b. stuck によるプログラムの実行例

```
push: 1, pop: 2
1
Input key
11
push: 1, pop: 2
1
Input key
12
push: 1, pop: 2
1
Input key
13
push: 1, pop: 2
1
Input key
14
push: 1, pop: 2
1
Input key
15
push: 1, pop: 2
2
key: 15
push: 1, pop: 2
2
key: 14
push: 1, pop: 2
2
key: 13
push: 1, pop: 2
2
key: 12
push: 1, pop: 2
2
key: 11
push: 1, pop: 2
1
Input key
16
push: 1, pop: 2
2
key: 16
push: 1, pop: 2
2
Stack is empty!!
push: 1, pop: 2
```

2-a.queue によるプログラム(MAX==5)

```
01 #include <stdio.h>
02 #define MAX 5 /*MAXの値を5と定義*/
03 int q[MAX],top,rear;
04
05 int main(){
06     int n,c;
07     initqueue(); /*関数 initqueue の呼び出し*/
08     printf("enqueue: 1, dequeue: 2\n");
09     while(scanf("%d",&c)!=EOF){ /*while 文,入力された値が EOF ではない間ループする*/
10         switch(c){ /*switch-case 文*/
11             case 1: /*変数 c の値が1の場合*/
12                 if(queue_is_full()==1){ /*if 文,関数 queue_is_fill を呼び出し,*/
13                     printf("Queue is full!!\n"); /*返された値が1の場合関数 printf の出力.*/
14                     break; /*if 文の処理が終わると break*/
15                 }
16                 printf("Input key\n"); /*追加する値の入力を促す */
17                 scanf("%d",&n); /*関数 scanf で入力された値を変数 n に代入し,*/
18                 enqueue(n); /*変数 n を引数として関数 enqueue を呼び出し,*/
19                 break; /*一通り処理を終えると break */
20             case 2: /*変数 c の値が2の場合*/
21                 if(queue_is_empty()==1){ /*if 文,関数 queue_is_empty を呼び出し, */
22                     printf("Queue is empty!!\n"); /*返された値が1の場合関数 printf の出力.*/
23                     break; /*if 文の処理が終わると break */
24                 }
25                 printf("key: %d\n",dequeue()); /*if 文の関数 queue_is_empty を呼び出し, */
26                 break; /*返された値が1でない場合, */
27             } /*関数 dequeue の出力.(関数 dequeue により */
28             /*返された値を整数として出力).処理を終えると break*/
29     printf("enpueue: 1, dequeue: 2\n");
30 }
31 }
32
33 int queue_is_empty(){ /*関数 queue_is_empty*/
34     if((rear-top)==0) return(1); /*if 文.rear-top の値が0の場合,値1を返す */
35     return(0); /*rear-top の値が0ではない場合,値0を返す*/
36 }
37
38 int queue_is_full(){ /*関数 queue_is_empty*/
39     if((rear-top) >= MAX) return(1); /*if 文.rear-top の値がMAX以上の場合,値1を返す*/
40     return(0); /*rear-top の値がMAX以上ではない場合,値0を返す*/
41 }
42
43 initqueue(){ /*関数 initqueue*/
44     top = rear = 0; /*top の値に0を代入*/
45 }
46
47 enqueue(int n){ /*関数 enqueue*/
48     q[(rear+MAX)%MAX]=n; /*引数として受け取った変数 n の値を代入する配列 q の */
49     rear++; /*添字を(rear+MAX)%MAX とすることで配列の要素を環状的に*/
50 } /*指定し,rear の値をインクリメント */
51
52 int dequeue(){ /*関数 dequeue*/
53     int key; /*配列 q の添字を(top+MAX)%MAX にする事ことで取り出す要素を*/
54     key=q[(top+MAX)%MAX]; /*環状的に値を指定し,key に代入する */
55     top++; /*top の値をインクリメント */
56     return(key); /*変数 key を返す */
57 }
```

2-b.queue によるプログラムの実行例

```
enqueue: 1, dequeue: 2
1
Input key
11
enqueue: 1, dequeue: 2
1
Input key
12
enqueue: 1, dequeue: 2
1
Input key
13
enqueue: 1, dequeue: 2
1
Input key
14
enqueue: 1, dequeue: 2
1
Input key
15
enqueue: 1, dequeue: 2
1
Queue is full!!
enqueue: 1, dequeue: 2
2
key: 11
enqueue: 1, dequeue: 2
2
key: 12
enqueue: 1, dequeue: 2
2
key: 13
enqueue: 1, dequeue: 2
2
key: 14
enqueue: 1, dequeue: 2
2
key: 15
enqueue: 1, dequeue: 2
2
Queue is empty!!
enqueue: 1, dequeue: 2
1
Input key
16
enqueue: 1, dequeue: 2
2
Input key
16
enqueue: 1, dequeue: 2
```

追記

stuck によるプログラムの作成は以外とすんなりいけましたが、queue によるプログラム作成では配列を環状的にするという処理に苦戦し、除算的に配列を指定することで環状的になることに気づいたときには、とてもすっきりしました。