

CS50 코칭스터디 2기



좋아요 6608 수강생 1342

CS50 코칭스터디란? 🤖

CS50 2기 공지글 📄

CS50 2기 코치단 👤

2기 부스터 소통공간 🗨️ 📺

개인 미션 공간 🗨️

2기 리드부스터 소통공간 🗨️

2기 부스터 문의사항 🗨️

⚠️ 학습 인증 안내

▼ 미션공개&라이브특강 ▼

0주차_오리엔테이션

1주차_컴퓨팅사고

2주차_C언어

3주차_배열

❤ 부스터 여러분~ 설레는 3주차 (1월 25일 ~ 1월 29일) 학습 시작해볼까요? ❤

○ 🤖 배열

○ 💎 3주차 미션! (1월 28일 목요일 오...

4주차_알고리즘

5주차_메모리

6주차_자료구조

🕒 코딩은 랜선을 타고

💎 3주차 미션! (1월 28일 목요일 오후 6시까지 제출)

📅 3주차 미션

💡 샘플미션 (제출용X)

▶ 3주차 부터는 부스터 여러분들께서 미션을 수행하시는데 도움이 될 수 있게 미션샘플을 답지와 함께 제공해드립니다. 미션샘플 출용이 아닌 아래의 미션을 풀기위해 참고하기 위한 미션입니다.

✓ 샘플미션. 채점 프로그램을 만들어보자!

1. 미션 제목

- 채점 프로그램을 만들어보자!

2. 지시문

학생들의 답안을 채점하는 프로그램을 작성하시오.

- 모든 문제는 객관식으로 1 ~ 5 사이의 정답을 가짐

- 10 문제에 대한 학생의 답을 매개변수 (arguments) 를 이용하여 입력

- 문제의 정답은 배열 (arrays) 를 이용하여 초기화 (아래 문제 정답 참조)

- 답안의 유효성을 체크 (답안의 개수: 10, 답의 범위: 1 ~ 5), 맞지 않으면 오류를 출력

- 답안의 개수가 틀린 경우, "문제는 10 문제입니다. 현재 {n} 개의 답안을 제출했습니다. 10 개의 답안을 제출하시오."

- 답의 범위가 틀린 경우, "답의 범위는 1 ~ 5 입니다. 범위 외의 답안이 있습니다."

- 문제당 10점으로 채점하고, 학점은 아래 "학점" 을 참고

- 정상 출력은 정답 수와 점수를 출력하고, 학점도 같이 출력

Table 1 - 학점

A+	A	B+	B	C+	C	D+	D	F
100~95	94~90	89~85	84~80	79~75	74~70	69~65	64~60	59~

유효성 체크 1: 답안 수 == 10

유효성 체크 2: 1 <= 답 <= 5

문제 정답 예시: 1, 2, 3, 4, 5, 1, 2, 3, 4, 5

3. 핵심 개념

#배열 #문기문 #반복문 #매개변수 #표준출력 #atoi #stdlib #break

4. 부가 설명

- char to int (atoi: stdlib.h): <https://ko.wikipedia.org/wiki/Stdlib.h>- 표준입출력: https://www.tutorialspoint.com/cprogramming/c_input_output.htm- break: https://www.tutorialspoint.com/cprogramming/c_break_statement.htm- malloc: <http://www.cplusplus.com/reference/cstdlib/malloc/>- strlen: <http://www.cplusplus.com/reference/cstring/strlen/>

🔔 답안

1 #include <stdio.h>

2 #include <stdlib.h>

3 #include <string.h>

4

5 const int NUMBER_OF_ANSWERS = 10;

6

수강완료

Alice코치 (team1~10)

L 코치질문 🗨

L 미션제출 📄

수료증

성적조회

```

9  const int SCORES[NUMBER_OF_GRADES] = {95, 90, 85, 80, 75, 70, 65, 60, 0};
10  const char *GRADES[NUMBER_OF_GRADES] = {"A+", "A", "B+", "B", "C+", "C", "D+", "D", "F"};
11
12  int* charArrayToIntArray(char *target[], const int length);
13  int getNumberOfCollectAnswers(const int answers[], const int target[], int length);
14  int calculateScore(int numberOfCollectAnswers);
15  void printAnswers(char *who, const int target[], int length);
16  char* calculateGrade(int totalScore, const int scores[], const char *grades[], int length);
17
18  int main(int argc, char *argv[]) {
19      if (argc != (NUMBER_OF_ANSWERS + 1)) {
20          printf("문제는 10 문제입니다. 현재 %d 개의 답안을 제출했습니다. 10개의 답안을 제출하시오.\n", argc - 1);
21          printf("(ex) ./mission02 4 4 3 5 2 5 1 2 4 3\n");
22          return -1;
23      }
24
25      int *studentAnswers = charArrayToIntArray(argv, argc);
26
27      for (int i = 0; i < NUMBER_OF_ANSWERS; i++) {
28          if (studentAnswers[i] >= 0 && studentAnswers[i] <= 5) {
29              continue;
30          } else {
31              printf("답의 범위는 1 ~ 5 입니다. 범위 외의 답안이 있습니다.\n");
32              return -1;
33          }
34      }
35
36      printf("학점평가 시작\n");
37
38      int numberOfCollectAnswers = getNumberOfCollectAnswers(ANSWERS, studentAnswers, NUMBER_OF_ANSWERS);
39
40      int totalScore = calculateScore(numberOfCollectAnswers);
41
42      char *grade = calculateGrade(totalScore, SCORES, GRADES, NUMBER_OF_GRADES);
43
44      printAnswers("정답", ANSWERS, NUMBER_OF_ANSWERS);
45      printAnswers("학생", studentAnswers, NUMBER_OF_ANSWERS);
46      printf("정답 수: %d\n", numberOfCollectAnswers);
47      printf("점수: %d\n", totalScore);
48      printf("학점: %s\n", grade);
49
50      return 0;
51  }
52
53  int* charArrayToIntArray(char *target[], const int length) {
54      int *result = malloc(sizeof(int) * NUMBER_OF_ANSWERS);
55
56      for (int i = 0; i < NUMBER_OF_ANSWERS; i++) {
57          result[i] = atoi(target[i + 1]);
58      }
59
60      return result;
61  }
62
63  int getNumberOfCollectAnswers(const int answers[], const int target[], int length) {
64      int numberOfCollectAnswers = 0;
65
66      for (int i = 0; i < length; i++) {
67          if (answers[i] == target[i]) {
68              numberOfCollectAnswers++;
69          }
70      }

```

<

>

수강완료

```
74
75     return numberOfCollectAnswers;
76 }
77
78 int calculateScore(int numberOfCollectAnswers) {
79     return numberOfCollectAnswers * 10;
80 }
81
82 void printAnswers(char *who, const int target[], int length) {
83     printf("%s: ", who);
84
85     for (int i = 0; i < length; i++) {
86         printf("%d\t", target[i]);
87     }
88 }
89
90 char* calculateGrade(int totalScore, const int scores[], const char *grades[], int length) {
91     char *grade;
92
93     for (int i = 0; i < length; i++) {
94         if (totalScore >= scores[i]) {
95             grade = malloc(sizeof(char) * strlen(grades[i]));
96             strcpy(grade, grades[i]);
97             break;
98         }
99     }
100
101     return grade;
102 }
```

✓ 문제 1. 학점을 계산해보자!

1. 미션 제목
학점을 계산해보자!

2. 지시문
- 학생의 점수로 학점을 구하는 프로그램을 작성하시오.
 - 키보드에서 입력받은 성적 (0 ~ 100 점) 의 유효성을 체크
 - 학점은 배열을 이용하여 초기화 (아래 "학점 테이블" 참조)
 - 학점은 "학점" 과 같이 계산하는데, 반드시 "학점 테이블"을 사용하여 계산하고 학점도 "학점 테이블"의 내용을 출력
 - 성적을 입력하여 계속 학점을 구하며 특별한 문자인 "999" 를 입력하면 프로그램을 종료

Table 1 - 학점

A+	A	B+	B	C+	C	D+	D	F
100~95	94~90	89~85	84~80	79~75	74~70	69~65	64~60	59~

Table 2 - 학점 테이블

Index	0	1	2	3	4	5	6	7	
학점	A+	A	B+	B	C+	C	D+	D	
성적	95	90	85	80	75	70	65	60	

유효성 체크: 0 <= 성적 <= 100
- "120" 입력 -> 성적을 올바르게 입력하세요! (0 ~ 100)

Figure 1 출력 결과

```
~/Documents/git-projects/boost-course ./mission01 4102 22:26
학점 프로그램
종료를 원하면 "999" 를 입력
[학점 테이블]
점수 : 95      90      85      80      75      70      65      60      0
학점 : A+      A       B+      B       C+      C       D+      D       F
성적을 입력하세요 (0 ~ 100) : 0
학점은 F 입니다.
성적을 입력하세요 (0 ~ 100) : 10
학점은 F 입니다.
성적을 입력하세요 (0 ~ 100) : 20
학점은 F 입니다.
성적을 입력하세요 (0 ~ 100) : 30
학점은 F 입니다.
성적을 입력하세요 (0 ~ 100) : 40
학점은 F 입니다.
성적을 입력하세요 (0 ~ 100) : 50
학점은 F 입니다.
성적을 입력하세요 (0 ~ 100) : 60
학점은 D 입니다.
성적을 입력하세요 (0 ~ 100) : 70
학점은 C 입니다.
성적을 입력하세요 (0 ~ 100) : 76
학점은 C+ 입니다.
성적을 입력하세요 (0 ~ 100) : 80
학점은 B 입니다.
성적을 입력하세요 (0 ~ 100) : 85
학점은 B+ 입니다.
성적을 입력하세요 (0 ~ 100) : 89
학점은 B+ 입니다.
성적을 입력하세요 (0 ~ 100) : 90
학점은 A 입니다.
성적을 입력하세요 (0 ~ 100) : 100
학점은 A+ 입니다.
성적을 입력하세요 (0 ~ 100) : 99
학점은 A+ 입니다.
성적을 입력하세요 (0 ~ 100) : 99
학점은 A+ 입니다.
성적을 입력하세요 (0 ~ 100) : 123
** 123 성적을 올바르게 입력하세요. 범위는 0 ~ 100 입니다.
성적을 입력하세요 (0 ~ 100) : -1
** -1 성적을 올바르게 입력하세요. 범위는 0 ~ 100 입니다.
성적을 입력하세요 (0 ~ 100) : 999
학점 프로그램을 종료합니다.
```

3. 핵심 개념

#배열 #표준입력 #표준출력 #분기문 #반복문 #break #무한반복문

4. 부가 설명

- 표준입출력: https://www.tutorialspoint.com/cprogramming/c_input_output.htm
- break: https://www.tutorialspoint.com/cprogramming/c_break_statement.htm

✓ 문제 2. 누가 빠졌는지 찾아보자!

1. 미션 제목

누가 빠졌는지 찾아보자!

2. 지시문

- 1 부터 N 까지의 숫자 모음이 있다. 여기서 K 라는 숫자가 빠진 N - 1 개의 파일이 있다. K 라는 숫자를 찾아보자.
 - N 은 2 이상 500,000 이하의 값을 가짐
 - 정렬되지 않은 숫자들의 모음 파일이 주어짐 (ex, 10.txt 100.txt 1_000.txt 10_000.txt 100_000.txt, 500_000.txt)
 - 위 파일에서 빠진 숫자 K 를 찾아라
 - 파일의 값을 읽어 n 과 k 가 빠진 arr 이를 저장하는 참고 코드는 아래 참고 (파일은 제공된 파일 사용)
 - 파일 구조: 첫 줄에는 N 값이 주어지고, 그 아래 줄에는 공백으로 K 를 제외한 N - 1 개의 숫자들이 나열 됨

```
1 #include <stdio.h>
2 #define SIZE 500000
3
4 int main(int argc, char*argv[]) {
5     int n;
```

```

9
10 // 1부터 N의 숫자중 K가 빠진 배열
11 int partArr[SIZE];
12 int lengthOfPartArr = n-1;
13
14 for(int i=0; i < lengthOfPartArr; i++){
15     scanf("%d", &partArr[i]);
16 }
17
18 // TODO: n과 partArr를 이용하여, K를 구하라
19 return 0;
}

```

정렬되지 않은 숫자들의 모음 파일

ZIP

Figure 1 - 실행 예시 및 정답

```

~/Documents/git-projects/boost-course ➤ ./mission03 < 10.txt
K = 7
~/Documents/git-projects/boost-course ➤ ./mission03 < 100.txt
K = 57
~/Documents/git-projects/boost-course ➤ ./mission03 < 1_000.txt
K = 507
~/Documents/git-projects/boost-course ➤ ./mission03 < 10_000.txt
K = 8072
~/Documents/git-projects/boost-course ➤ ./mission03 < 100_000.txt
K = 96676
~/Documents/git-projects/boost-course ➤ ./mission03 < 500_000.txt
K = 264936
~/Documents/git-projects/boost-course ➤

```

```

10.txt: k = 7
100.txt: k = 57
1_000.txt: k = 507
10_000.txt: k = 8072
100_000.txt: k = 96676
500_000.txt: k = 264936

```

3. 핵심 개념

#배열 #정수합 #합의공식 #반복문

4. 부가 설명

- 표준입출력: https://www.tutorialspoint.com/cprogramming/c_input_output.htm
- https://en.wikipedia.org/wiki/1_%2B_2_%2B_3_%2B_4_%2B_%E2%8B%AF

✓ 문제 3. Queue를 만들어보자!

1. 미션 제목

Queue 를 만들어 보자!

2. 지시문

배열을 이용하여 Queue 를 만들어 보자.

특정 업무를 할 때, 우리는 일을 들어온 순서대로 해야할 때가 있다. 은행 업무를 예를 들어보자. 은행업무를 보기 위한 고객들이 있다고 치고, 각자 대기표가 있다. 그럼 은행원들은 각자 업무가 끝나면 대기한 고객을 순서대로 뽑아야 할 것이다. 이때 필요한 자료구조는 Queue 이다. (1) 대기표를 뽑는다 (Queue 에 데이터를 삽입). (2) 대기인원을 보여준다 (queue 에 쌓여있는 데이터 조회). (3) 순서기인원을 호출한다 (queue 를 하나씩 pop 한다).

- Queue 자료구조를 array를 이용해 구현

1. add (1). pop (2). display (3). quit (4) 기능 구현



수강완료

- queue 가 꽉찼는지 확인 (Queue 의 max 크기는 10으로 정의), queue 가 꽉찼으면 "Queue가 꽉 찼습니다." 를 출력
 - queue 에 삽입이 가능하면, 값을 입력 받아 queue 배열에 삽입 (hint: front, rear 변수를 사용하여 queue 의 현재 위치를 다)
5. pop() 함수의 정의
- queue 가 비었는지 확인, 비었으면 "Queue가 비었습니다." 를 출력
 - queue 가 비어있지 않으면, 가장 먼저 들어온 순서로 값을 하나 가져와 출력 (hint: front 변수값 조정 필요)
6. display() 함수의 정의
- 반복문을 사용하여 배열의 모든 요소를 출력 (hint: front, rear 변수 범위로 배열값을 출력)

TODO: 여기서 출력 예시 보여주기

```

입력 : 2
큐 가 비어 있음
1. 삽입
2. 삭제
3. 조회
4. 종료
입력 : 1
삽입할 값 : 10
1. 삽입
2. 삭제
3. 조회
4. 종료
입력 : 1
삽입할 값 : 20
1. 삽입
2. 삭제
3. 조회
4. 종료
입력 : 1
삽입할 값 : 13
1. 삽입
2. 삭제
3. 조회
4. 종료
입력 : 1
삽입할 값 : 1
1. 삽입
2. 삭제
3. 조회
4. 종료
입력 : 3
큐 : 10 20 13 1
1. 삽입
2. 삭제
3. 조회
4. 종료
입력 : 2
큐에서 삭제됨 : 10
1. 삽입
2. 삭제
3. 조회
4. 종료
입력 : 1
삽입할 값 : 40
1. 삽입
2. 삭제
3. 조회
4. 종료
입력 : 3
큐 : 20 13 1 40
1. 삽입

```

3. 핵심 개념 (키워드 제시)

#배열 #Queue #switch문 #반복문 #표준입출력

4. 부가 설명 (만약 존재한다면)

- Queue: [https://en.wikipedia.org/wiki/Queue_\(abstract_data_type\)](https://en.wikipedia.org/wiki/Queue_(abstract_data_type))
- switch: <https://docs.microsoft.com/ko-kr/cpp/c-language/switch-statement-c?view=vs-2019>
- 표준입출력: https://www.tutorialspoint.com/cprogramming/c_input_output.htm

*** 미션(3문제) 중에 회의를 통해 1개를 선택해서, 답을 목요일에 리드부스터님이 팀별로 1개만 지면 됩니다.**

💡 문제를 풀다가 도저히 풀 수 없을때는 어떻게 하나요?

🗣 팀 내 해결이 안되는 문제 혹은 구글링으로도 이해가 어려운 질문사항은 왼쪽의 "코치질문" 게시판에 질문해주세요!

여러분들의 담당 코치가 답변을 해줄 것이나 해당 질문에 답을 아는 동료부스터 누구나 답변을 남길 수 있으며 C 코칭스터디는 동료피드백을 통한 서로의 성장을 응원합니다. ♥

💡 우수 미션으로 뽑히고 싶은데 기준이 어떻게 되나요?

🗣 제출한 미션에 우수답변으로 문항 별 2~3개정도 선별하여 매주 금요일 라이브 특강에서 소개합니다. 우수미션 선정 기준은 다음과 같습니다.

- 1)코드의 정확성,효율성 2) 아이디어의 참신함, 창의성 3) 다양한 방법으로의 시도

💡 제출한 미션에 대한 피드백은 언제 주어나요?

🗣 제출한 미션에 대한 피드백은 다음날 금요일 정오 전까지 여러분의 담당 코치가 제공합니다.

또한 매주 금요일 8시 진행되는 라이브 특강을 통해 다른 팀의 미션 해결 사고방법과 다른 부스터들의 창의적인 소개할 예정입니다.

공유하기

좋아요 82

댓글 16

연관 토론 223

댓글을 입력하세요.

☐ 비공개

6조kigga 2021.01.31 23:22 신고

2주차와 같은 3주차 미션해설영상은 따로 안올려주시나요?

추천 2

답글



4조수달이 2021.01.27 23:21 신고



수강완료

추천 12

답글

**부스트코스 관리자** 2021.01.27 22:54 신고

안녕하세요! 부스트코스 운영진입니다.

3주차 미션 난이도에 대한 많은 의견들이 있으신데요!

중요한 내용들, 꼭 필요한 내용들을 모두 담아내려고 문제를 출제하다보니 난이도에 대한 이슈가 있는 것 같습니다.

4주차부터는 여러분의 많은 의견을 반영해 '힌트'를 도입하고 더욱 **다양한 난이도**의 문제를 조절하여 출제하도록 하

다.

미션 풀이도 물론 중요하지만, 여태까지 코치님의 LIVE에서도 보셨다시피 개발에서는 협업이 정말 중요합니다.

혹시 모르는 부분이 있으시다면 팀원분들에게 질문을 하고 내가 아는 부분이면 친절하게 알려줘서 이번 스터디가 마

업에서 프로젝트를 하는 느낌을 받으셨으면 좋겠습니다.

완벽한 코드가 아니어도 좋습니다! C언어가 아니어도 괜찮습니다! 의사코드만으로 작성해주셔도 괜찮습니다!

우리는 이제 첫 발을 내디뎠는걸요! :) 이번주의 큰 산만 넘어주세요!

별써 스터디의 절반까지 잘 달려오셨는데, 정말 고생이 많으시고 힘드시겠지만 다시 처음에 마음을 다잡고 화이팅 하

원합니다.💪💪

운영진도 최대한 도움을 드릴 수 있도록 노력하겠습니다!!

감사합니다.

추천 10

답글 1

2조알파카짱 2021.01.27 22:20 신고

함께 해결하는 팀미션이어도 어느 정도는 아는게 있어야 의견 교환을 하거나 구글링을 할 수 있는건데 cs를 이번에 입

람에게는 미션이 너무 어렵습니다. 샘플문제와 답안을 주셨지만 갑자기 어떻게 저런 답안이 나오게 되었는지 이해가 ;

고 그 답안이 어떻게 미션을 푸는데 도움이 되는건지도 아예 감이 안잡힙니다. 질문 받아주시는 코치님들이 계시지만

결을 위해 뭘 질문해야 하는건지도 아예 모르겠습니다..

추천 15

답글

**2조부르주아** 2021.01.27 21:18 신고

강의만 듣다가 미션내용보고 깜짝놀랐습니다. 부가자료 첨부는 되어있으나 영어사이트이고.. 아직 배우지도 않은 포인

텀이나와서 샘플로 주신 답변도 흐름은 이해가 가지만 정확한 처리 과정은 이해되지않습니다. 예전에 잠시 포인터에

배운적이 있음에도 갑자기 난이도가 확 뛰어 버린 것 같습니다. 포인터 개념은 컴공과 학생들도 어려워 하는 개념으로

있는데 비전공자가 갑자기 응용을 하기에는 뭔가 강의수준과 미션과의 괴리감이 크고 미션에 대한 부가적 정보도 부족

같아 아쉽습니다.

추천 19

답글

**3조거부기** 2021.01.27 21:07 신고

미션이 어려우신 분들은 샘플미션의 코드 한줄한줄 이해하려고 하지 않으셔도 된다는 말을 해드리고 싶네요

진도상 아직 안배운것들이 나오고 아래의 미션을 해결하는데 필수인 부분도 아닙니다.

그보단 샘플미션 코드에서 함수들이 어떤순서로 실행되는지 그 흐름을 보시는게

아래 미션을 해결하는데 도움이 될거라고 생각합니다.

함수이름이 직관적이니 각 함수가 어떤 역할인지는 이름으로 유추하시면 되고,

이런식으로 한 줄씩해서 실행되는 과정을 보시면 분명에 해결방법도 있으실거라 생각합니다.

<

>

수강완료

지금까지 배운내용으로 충분히 해결할 수 있는 수준이에요 그리고 약간 어려운걸 풀어내야 성장하는게 사실이구요. 물론 약간이라는 점에서 동의하지 않는 분이 많을것 같습니다만, 프로그래밍에 처음 접하기 때문이라고 생각해요. 어떻게 접근해야할지 모르겠어서 골머리 썩는게 필수로 거쳐갈 수 밖에 없는 과정 중의 하나예요...

무엇보다 운영진께서 언급하셨듯 질문이 중요한데,
다만 관리가 용이하기 위해 5명 내외의 팀으로 분류를 나뉘기 때문에 아쉬웠어요.

좀 더 많은 인원이 모여서 함께 이야기해야 질문과 답변이 더 원활하게 오간다고 생각해서요
그래서 부스터 소통공간 게시판에도 작성했지만 질문과 답이 오가는 공간을 개인적으로 만들었으니
관심이 있으시면 들어오셔서 질문하고 같이 고민하면 도움이 될 거라 생각합니다.

https://join.slack.com/t/bc-7u48203/shared_invite/zt-l66gv0rq-qGSsVKzKZ_T_DowmbMQttqQ

추천 8

답글



1조이모지 2021.01.27 17:22 신고

제가 개인적으로 포인트 변수 너무 어려워서 ㅈㅈ 학습자료 혼자서 만들어봤는데 이미 저희 팀 내에서는 공유했지만 ㅈㅈ에서 혹시 필요하신 분들 계시다면 참고해주세요!! ☺ 다들 화이팅입니다 ㅈㅈ (혼자 공부하면서 만든 거라 구어체가 많이 사용되었습니다)

3주차 포인터 학습.pdf

저장

추천 41

답글 9



8조세바스찬 2021.01.27 15:47 신고

강의 내용에 비해 미션이 너무 어렵습니다. 몇시간째 고민해봐도 어떻게 풀어야 할지 답이 안나와요. 샘플미션 답안도 부분이 많구요. 아무리 팀미션이라도 강의 수준에 맞는 미션을 주셨으면 좋겠습니다. 배우는 재미를 느끼기보다 큰 장끼네요.

추천 41

답글



Holly 2조리드홀리 2021.01.27 05:40 신고

문제의 난이도가 너무 갑자기 높아진 것 같습니다. 라이브때 미리 말씀해주시긴 하셨지만.. CS에 기본적인 지식이 없는게는 너무 난이도가 높아서 어떻게 문제를 풀어야 할지 막막하네요. 팀별 과제이긴 하지만 난이도를 좀 조절해주시면 의견 남겨요~

추천 42

답글 3

7조리드CoderK 2021.01.26 22:34 신고

2번문제 부가설명 <redirection을 이용한 파일입력>링크가 필요한데 어떻게하죠?

추천

답글



10조녹용검 2021.01.26 16:53 신고

3번 문제는 데이터 삽입을 10번까지만 할 수 있는 큐를 만드는 건가요?
아니면 지우고 삽입하는 동작을 꼭차지 않는 이상 무한정 할 수 있는 큐를 만드는 건가요

추천

답글



수강완료



8조슈슈 2021.01.26 16:13 신고

1번 예시문을 그대로 복사해서 실행시켜보니 에러가 나는데요....

추천 21 답글 2



7조ylem76 2021.01.25 23:35 신고

3번 문제에 TODO: 여기서 출력 예시 보여주기라고 쓰여있는데, 따로 첨부된 예시가 없는 게 맞는 건가요?

추천 답글 1



7조리드내일은코딩왕 2021.01.25 22:27 신고

2번문제 부가설명 <redirection을 이용한 파일입력>의 링크로 이어지는 페이지가 나오지 않습니다. 다른 링크 부탁드

추천 1 답글 1



7조yoon 2021.01.25 21:56 신고

문제1. 지시문 5번째줄에는 -1 입력시 종료라고 되어있는데 출력화면에는 999를 입력해야 종료되는걸로 되어 있습니다 어디에 맞추면 될까요?

추천 1 답글 1



7조맹범 2021.01.25 19:03 신고

미션 프로그램 다 작성하고 문서화했는데 미션이 수정돼버리다니ㅠㅠㅠㅠ

추천 2 답글 1

☐ 1 ☐