

5월 30일 실습강의

C프로그래밍

실습 review

실습 22

- 두 숫자의 대소 관계를 비교하는 프로그램을 작성하시오.
- 함수 void *min(void *ptr1, void *ptr2, char option)을 구현
 - ptr1이 참조하는 숫자와 ptr2가 참조하는 숫자의 대소 관계를 비교하여 더 작은 수에 대한 포인터를 반환하는 함수
 - option은 비교 대상이 되는 숫자의 type을 의미
 - option == 'i'인 경우 두 숫자를 int로 간주
 - option == 'f'인 경우 두 숫자를 float으로 간주
 - 위의 두 경우를 제외하고는 고려하지 않아도 됨
 - 적절한 casting이 필요
- 위 함수를 이용하여 작은 수를 출력

- 출력 예시

```
gr120200190@cspro:~$ ./runs/17
Enter the type of numbers: i
Enter the 1st number: 123
Enter the 2nd number: 321

minimun value is 123
gr120200190@cspro:~$ ./runs/17
Enter the type of numbers: f
Enter the 1st number: 1.23
Enter the 2nd number: 3.21

minimun value is 1.230000
```

실습 22

데이터 타입에 맞게 입력

```
18 int main(void){
19     char option;
20     void *ptr1, *ptr2, *res;
21     int i1, i2;
22     float f1, f2;
23
24     printf("Enter the type of numbers: ");
25     scanf("%c", &option);
26
27     if(option == 'i'){
28         printf("Enter the 1st number: ");
29         scanf("%d", &i1);
30         printf("Enter the 2nd number: ");
31         scanf("%d", &i2);
32
33         ptr1 = &i1;
34         ptr2 = &i2;
35     }
36     else if(option == 'f'){
37         printf("Enter the 1st number: ");
38         scanf("%f", &f1);
39         printf("Enter the 2nd number: ");
40         scanf("%f", &f2);
41
42         ptr1 = &f1;
43         ptr2 = &f2;
44     }
45
46     res = min(ptr1, ptr2, option);
47
48     if(option == 'i')
49         printf("\nminimum value is %d\n", *(int*)res);
50     else if(option == 'f')
51         printf("\nminimum value is %f\n", *(float*)res);
52
53     return 0;
54 }
```

```
1 #include <stdio.h>
2
3 void *min(void *ptr1, void *ptr2, char option){
4     if(option == 'i'){
5         if(*(int*)ptr1 < *(int*)ptr2)
6             return ptr1;
7         else
8             return ptr2;
9     }
10    else if(option == 'f'){
11        if(*(float*)ptr1 < *(float*)ptr2)
12            return ptr1;
13        else
14            return ptr2;
15    }
16 }
```

데이터 타입에 맞게 형변환

C프로그래밍 실습

실습 23

- 두 개의 문자열을 입력 받아 하나의 문자열로 붙이는(concatenate) 프로그램을 작성하시오.
 - 입력 받을 문자열의 크기를 먼저 입력
 - 해당 크기 만큼의 문자열을 입력
 - 입력 받은 두 문자열을 순서대로 붙여 출력
 - 문자열을 이어 붙여주는 아래 함수를 작성
 - `void concat(char *str, char *str1, char *str2, int len1, int len2);`
 - `str1`과 `str2`를 붙여 `str`에 저장해주는 함수
 - `string.h` 라이브러리는 사용 금지
 - 각각의 문자열은 최대 10자리로 가정
 - 생각보다 많은 에러를 만나게 될 것 입니다...
(입출력, 메모리, `char*`)
(hint: `getchar();`)

- 출력 예시

```
Input size of str1: 17
Input str1: C is really fun.
Input size of str2: 8
Input str2: Is it..?
result: C is really fun. Is it..?
```

실습 24

- 두 개의 3x3 행렬의 뺄셈을 수행하는 프로그램을 작성하시오.
- 원소가 실수인 두 개의 3x3 행렬을 입력
- 먼저 입력 받은 행렬에서 나중에 입력 받은 행렬을 뺀 결과를 출력

| | | | | | | | | | | |
|-----|-----|-----|---|-----|-----|-----|---|------|------|------|
| 1.4 | 9.8 | 2.5 | | 7 | 4.8 | 4.5 | | -5.6 | 5 | -2 |
| 1.2 | 2.1 | 8.7 | — | 4.3 | 2.6 | 5.6 | = | -3.1 | -0.5 | 3.1 |
| 3.2 | 3.1 | 3.7 | | 1.3 | 4.1 | 8.4 | | 1.9 | -1 | -4.7 |

- 행렬의 뺄셈은 한 행 단위로 작업하는 별도의 함수를 작성

void sub_matrix(double *left_row, double *right_row, double *result_row, int n);

left 행렬의 한 행

right 행렬의 한 행

result 행렬의 한 행

전달받은 행의 크기

- 함수 내부에서는 index syntax 사용 금지 (left_row[i][j] 등)
오직 pointer 연산을 통해서 수행

- 출력 예시

```
Input first matrix:
```

```
1.4 9.8 2.5
```

```
1.2 2.1 8.7
```

```
3.2 3.1 3.7
```

```
Input second matrix:
```

```
7 4.8 4.5
```

```
4.3 2.6 5.6
```

```
1.3 4.1 8.4
```

```
Result:
```

```
-----
```

```
| -5.600 | 5.000 | -2.000 |
```

```
-----
```

```
| -3.100 | -0.500 | 3.100 |
```

```
-----
```

```
| 1.900 | -1.000 | -4.700 |
```

```
-----
```

실습 제출 양식

- 각 실습의 파일명은 {학번}_{실습번호}.c로 저장
e.g.) 실습6의 파일명은 20211234_6.c
- 작성한 c파일 코드들 (.c 파일들)을 사이버캠퍼스 과제란에 업로드
- 제출기한 : 사이버캠퍼스에 명시된 기한까지
- **지각제출은 불가능**
- **파일제목 등 제출 양식이 틀리면 오답처리**