고급 C프로그래밍 High Level C Programming

CHAPTER 3

배열의 이해 및 응용 실습

```
#include <stdio.h>
    #include <stdlib.h>
    void fill_2d_array(int(*parr)[5], int n);
    int main(void){ int arr[4][5] = { 0 };
    int* parr = arr;
    int num;
    printf("배열의 원소에 저장할 값? ");
   scanf("%d", &num); fill_2d_array(parr, num);
10
11 \cdot for (int i = 0; i < 4; i++) {
12 -
        for (int j = 0; j < 5; j++) {
             printf("%2d", arr[i][j]);
13
14
        printf("\n");
15
16
17 }
18
19
20 - void fill 2d array(int(*parr)[5], int n) {
         for (int i = 0; i < 4; i++)
21
22
             for (int j = 0; j < 5; j++)
23
                 parr[i][j] = n;
24
🐼 Microsoft Visual Studio 디버그 콘솔
                                               배열의 원소에 저장할 값? 5
55555
55555
55555
55555
C:\Users\KB\Documents\GitHub\hyun c\Projct\Project2\Debug\Project2.exe(프로세스 8112개)이(가) 종료되었습니다 .
```

```
2 #include <stdlib.h>
 3 void add_maxtrix(int(*parrX)[3], int(*parrY)[3]);
 4 - int main(void){
        int arrX[3][3] = \{ 10,20,30,40,50,60,70,80,90 \};
        int arrY[3][3] = \{9,8,7,6,5,4,3,2,1\};
        int* parrX = arrX;
        int* parrY = arrY;
        printf("x 행렬:\n");
        for (int i = 0; i < 3; i++) {
            for (int j = 0; j < 3; j++) {
                 printf("%5d", arrX[i][j]);
13
14
            printf("\n");
15
16
        printf("y 행렬:\n");
        for (int i = 0; i < 3; i++) {
18 -
             for (int j = 0; j < 3; j++) {
                                                               🐼 Microsoft Visual Studio 디버그 콘솔
                 printf("%5d", arrY[i][j]);
20
                                                                10 20 30
21
            printf("\n");
                                                                40 50 60
22
                                                                70 80 90
23
        printf("x+y 행렬:\n");
24
        add maxtrix(parrX, parrY);
                                                                   8
25 }
26 void add_maxtrix(int(*parrX)[3],int(*parrY)[3]) {
                                                              x+y 행렬:
27 -
        for (int i = 0; i < 3; i++) {
                                                               19 28 37
            for (int j = 0; j < 3; j++) {
                                                                46 55 64
29
                 printf("%5d", parrX[i][j] + parrY[i][j]);
                                                                73 82 91
31
            printf("\n");
                                                              C:\Users\KB\Documents\GitHub\hyun c\Projct\Project2\Debug\Project2.exe(프로세스 13932개)이(가) 종료되었습니
                                                              다(코드: 0개).
32
                                                              이 창을 닫으려면 아무 키나 누르세요...
33 }
```

```
1 #include <stdio.h>
3 int main() {
       int numarr[4][4]; // 4x4인 배열 생성
       int k = 0;// 줄바꿈을 위한 변수
       for (int i = 0; i < 4; i++) {
           for (int j = 0; j < 4; j++)
              scanf("%d", &numarr[i][j]);
11
       //출력
       printf("\n[변경 전]\n");
       for (int i = 0; i < 4; i++) {
                                                                      10
                                                                       11
           for (int j = 0; j < 4; j++) {
                                                                       12
               printf("%3d", numarr[i][j]);
17
                                                                       13
               k++;
                                                                       14
                                                                       15
               if (k % 4 == 0) printf("\n"); //4개印다 줄버꿈
                                                                       「변경 전 1
       printf("\n[변경 후]\n");
       for (int i = 0; i < 4; i++) {
           for (int j = 0; j < 4; j++) {
                                                                       [변경 후]
               printf("%3d", numarr[j][i]); // i <-> j 로 행렬 transpose
               k++;
              if (k % 4 == 0) printf("\n"); //4개印다 줄버꿈
       return 0;
32 }
```

```
9 10 11 12
13 14 15 16
     9 13
   7 11 15
   8 12 16
```

```
//행렬B 출력
                                                                  printf("[ 행렬 B ]\n");
int main() {
                                                                                                                              행렬
                                                                   for (i = 0; i < 3; i++) {
                                                                                                                                       A ]
    int arr1[3][3]; //≝ ∄ A
                                                                     for (j = 0; j < 3; j++) {
    int arr2[3][3]; //행렬 B
                                                                        printf("%4d", (*(*(ptr2 + i) + j)));
    int arr3[3][3]; //행렬 C (행렬 곱 결과 값 담을 행렬)
    int i, j, k;
    int(*ptr1)[3] = arr1; // arr1[i][j] == *(*(ptr1+i)+j
                                                                     printf("\n");
    int(*ptr2)[3] = arr2; // arr2[i][j] == *(*(ptr2+i)+j
    int(*ptr3)[3] = arr3; // arr3[i][j] == *(*(ptr3+i)+j
    printf("[ 행렬 A ]\n");
    for (i = 0; i < 3; i++) {
                                                                                                                              행렬 B ]
                                                                   for (i = 0; i < 3; i++) {
        for (j = 0; j < 3; j++) {
                                                                     for (j = 0; j < 3; j++) {
             scanf("%d", &(*(*(ptr1+i)+j)));
                                                                        arr3[i][j] = 0;
                                                                        for (k = 0; k < 3; k++) {
                                                                           *(*(ptr3 + i) + j) += (*(*(ptr1 + i) + k)) * (*(*(ptr2 + k) + j));
    //행렬B 입력받기
    printf("[ 행렬 B ]\n");
    for (i = 0; i < 3; i++) {
        for (j = 0; j < 3; j++) {
                                                                                                                              행 렬
            scanf("%d", &(*(*(ptr2+i)+j)));
                                                                                                                                        A ]
                                                                                                                                         2
                                                                                                                                1
                                                                                                                                                  3
                                                                   printf("[ 행렬 곱 ]\n");
                                                                                                                                         5
                                                                                                                                                  6
                                                                                                                                4
                                                                   for (i = 0; i < 3; i++) {
                                                                                                                                         8
                                                                                                                                                  9
                                                                     for (j = 0; j < 3; j++) {
                                                                                                                              행 렬
                                                                                                                                         B 1
                                                                                                                                9
                                                                                                                                         8
                                                                                                                                                  7
    printf("[ 행렬 A ]\n");
                                                                           ntf("%4d", arr3[i][j]);
                                                                                                                                6
                                                                                                                                         5
                                                                                                                                                  4
    for (i = 0; i < 3; i++) {
                                                                                                                                         2
                                                                                                                                3
                                                                                                                                                  1
        for (j = 0; j < 3; j++) {
                                                                    printf("\n");
                                                                                                                                                1
            printf("%4d", (*(*(ptr1 + i) + j)));
                                                                                                                              30
                                                                                                                                       24
                                                                                                                                               18
                                                                                                                                               54
                                                                                                                              84
                                                                                                                                       69
                                                                  return 0;
        printf("\n");
                                                                                                                            138 114
                                                                                                                                                90
```

```
void maxnum(int **pptr, int numarr[]); //함수 선언(최대값 구하는 함수)
  5 int main() {
        int numarr[5]; //배열 선언
        int max = 0; //최댓값을 받을 변수
        int* ptr = &max; // ptr = > max의 메모리 주소
        int** pptr = &ptr; // pptr = > ptr의 메모리 주소
        for (int i = 0; i < 5; i++) { //5개 수 입력 받기
           scanf("%d", &numarr[i]);
        maxnum(pptr, numarr); //함수 호출
        printf("\n최대값: %d\n", max);
        return 0;
 19 }
 21 void maxnum(int **pptr, int numarr[]) { //함수 정의
        for (int i = 0; i < 5; i++) {
           if(**pptr < numarr[i]) { //numarr[i]값이 max값보다 크면
               27 }
V 2 3
34
22
45
최대값 : 45
```

```
3 void main()
4 - {
      unsigned char limit_table[3] = { 4, 2, 3 };
      /* 연령별 윗몸 일으키기 횟수를 저장할 배열 */
      unsigned char count[3][4];
      int age, member, temp, sum;
      for (age = 0; age < 3; age++) {
         /* '20대 연력의 윗몸 일으키기 횟수'라고 출력하기 위해서 */
         printf("\n%d0대 연령의 윗몸 일으키기 횟수\n", age + 2);
         /* 해당 연령에 소속된 사람들을 순서대로 입력 받는다. */
         for (member = 0; member < limit_table[age]; member++) {</pre>
             printf("%dth : ", member + 1);
             scanf("%d", &temp);
             count[age][member] = (unsigned char)temp;
      printf("\n\n연령별 평균 윗몸 일으키기 횟수\n");
      /* 연령별로 입력된 횟수를 합산하여 평균 값을 출력한다 */
      for (age = 0; age < 3; age++) {
         sum = 0;
         printf("%d0대 : ", age + 2);
         for (member = 0; member < limit_table[age]; member++) {</pre>
             sum = sum + count[age][member];
         /* 합산값을 인원수로 나누어서 평균을 낸다.*/
         printf("%5.2f\n", (double)sum / limit_table[age]);
```

```
20대 연령의 윗몸 일으키기 횟수
1th : 57
2th : 52
3th : 55
4th: 50
30대 연령의 윗몸 일으키기 횟수
1th: 48
2th : 41
40대 연령의 윗몸 일으 🎾기 횟수
1th : 35
2th : 52
3th : 43
연령별 평균 윗몸 일으키기 횟수
20대 : 53.50
30대 : 44.50
40대 : 43.33
```

```
#include <stdio.h>
    int main(void)
 4 - {
         int ary[4][5];
         int \overline{i}, \overline{j}, \overline{cnt} = 1;
         for (i = 0; i < 4; i++)
             for (j = 0; j < 5; j++)
11 -
12
                  ary[i][j] = cnt++;
         for (i = 0; i < 4; i++)
             for (j = 0; j < 5; j++)
                  printf("%5d", ary[i][j]);
             printf("\n");
         return 0;
26 }
```

```
2
           3
                      5
           8
 6
                 9
                     10
11
     12
          13
                14
                     15
          18
                19
                     20
16
     17
```

```
4 int main(void)
  5 - {
       char fruit[5][20]; // 과일 이름을 입력할 배열
       int i, len;
       int max = 0;
       int index = 0;  // 길이가 가장 긴 과일 이름의 배열의 위치
       printf("5개의 과일 이름 입력 : ");
       for (i = 0; i < 5; i++)
          scanf("%s", fruit[i]); // 과일 이름 입력
       for (i = 0; i < 5; i++)
          len = strlen(fruit[i]); // 과일 이름의 길이 계산
          if (len > max)
             max = len; // 최대 길이로 하고
             index = i;  // 그 위치를 index에 기억
       printf("길이가 가장 긴 과일 이름은 %s입니다.", fruit[index]); // 출력
       return 0;
 29 }
Y 2 3
5개의 과일 이름 입력 : kiwi
banana
apple
strawberry
pineapple
길이가 가장 긴 과일 이��은 strawberry입니다.
```

```
3 int main(void)

      char word[10][21];
      // 최대 10개의 단어, 단어의 길이는 최대 20차

      char temp[21];
      // 단어 입력 시 사용할 임치 배월

      int cnt;
      // 입력된 단어의 수

        int i, j, dup, find;
                  tf("> 단어 입력 : ");
f("%s", temp);
             if(strcmp(temp, "end") == 0) break;
// 입력된 단어가 있는지 중복 여부 검사
             dup = 0;
             for(j = 0; j < i; j++)
                  if(strcmp(temp, word[i]) == 0)
                      dup = 1;
             if(dup == 1)
                      intf("# 이미 입력되었습니다!\n"); // 왜시지 출력하고
ntinue; // 배열에 넣는 작업을 건너 뜀
                   y(word[i], temp);
        printf("# 총 %d개의 단어가 입력되었습니다!\n\n", cnt);
            scanf("%s", temp);
if(strcmp(temp, "end") == 0) break;
             find = 0;
             for(i = 0; i < cnt; i++)</pre>
                  if(strcmp(temp, word[i]) == 0)
                      find = 1;
             if(find == 1)
                  printf("# %d번째 같은 단어가 있습니다!\n\n", i + 1);
                  printf("# 없는 단어입니다.\n\n");
```

```
> 단어 입력 : tiger
> 단어 입력 : apple
> 단어 입력 : pear
> 단어 입력 : eagle
> 단어 입력 : end
# 총 4개의 단어가 입력 (었습니다!
> 검색 단어 (종료는 end입력) : pear
# 3번째 같은 단어가 있 (기다!
> 검색 단어 (종료는 end입력) :
```

```
#include <stdio.h>
                                                                                  for(i = 0; i < 3; i++)
   #include <string.h>
                                                                                      for(j = i + 1; j < 4; j++)
    int main(void)
                                                                                         if(sp[i][6] < sp[j][6])</pre>
5 - {
        char branch[4][10] = {"관악점", "강남점", "명동점", "대림점"};<sub>41</sub>
                                                                                             stp = sp[i];
        int selling[4][7] = {
                                                                                             sp[i] = sp[j];
                                                                                             sp[j] = stp;
            {70, 45, 100, 92, 150, 81},
            {88, 92, 77, 30, 52, 55},
                                                                                             btp = bp[i];
            {50, 90, 88, 75, 77, 49},
                                                                                             bp[i] = bp[j];
            {120, 92, 80, 150, 130, 105}
                                                                                             bp[j] = btp;
12
        };
                                     > 실적별 출력...
        char *bp[4];
                                     대림점 120
                                                92
                                                      80 150 130 105
                                                                      113
        int *sp[4];
                                     관 악 점
                                             70
                                                 45
                                                     100
                                                          92 150
                                                                   81
                                                                       90
                                                                                  printf("> 실적별 출력...\n");
        int i, j, tot;
                                     명동점
                                             50
                                                 90
                                                      88
                                                                   49
                                                                                  for(i = 0; i < 4; i++)
        char *btp;
                                                 92
                                             88
                                                                        66
                                                                           53 -
        int *stp;
                                                                                      printf("%s", bp[i]);
                                     > 지점별 출력...
        for(i = 0; i < 4; i++)
                                                                                      for(j = 0; j < 7; j++)
                                                                        90 55
                                     관 악점
                                             70
                                                45 100
                                                          92 150
                                                                   81
                                     강남점
                                                 92
                                                          30
                                                              52
                                                                   55
                                                                        66 56
                                             88
            tot = 0;
                                                                                         printf("%5d", sp[i][j]);
                                     명동점
                                             50
                                                 90
                                                      88
                                                                   49
                                                                        72 57
            for(j = 0; j < 6; j++) 대림점 120 92
                                                      80 150 130 105
                                                                      113 58
                                                                           59
                                                                                      printf("\n");
                tot += selling[i][j];
                                                                                  printf("\n> 지점별 출력...\n");
                                                                                  for(i = 0; i < 4; i++)
            selling[i][6] = (int)(tot / 6.0 + 0.5);
                                                                                      printf("%s", branch[i]);
        for(i = 0; i < 4; i++)
                                                                                      for(j = 0; j < 7; j++)
                                                                                         printf("%5d", selling[i][j]);
            sp[i] = selling[i];
                                                                                      printf("\n");
        for(i = 0; i < 4; i++)
            bp[i] = branch[i];
```

```
#include <string.h>
   |int input_name(char (*name)[20]);
                                                        lint dup check(char (*name)[20], char *temp, int cnt)
   int dup_check(char (*name)[20], char *temp, int cnt);
                                                    43 - {
  void print_name(char (*name)[20], int cnt);
                                                            int i;
  int main(void)
                                                             for(i = 0; i < cnt; i++)
      char name[10][20];
                                                    47 -
      int cnt;
                                                                if(strcmp(name[i], temp) == 0) return 1;
      cnt = input_name(name);
      print_name(name, cnt);
                                                             return 0;
14
                                                    51 }
15 }
                                                        void print_name(char (*name)[20], int cnt)
16 int input name(char (*name)[20])
17 - {
                                                    53 → {
18
      char temp[20];
                                                            int i;
      int cnt = 0;
                                                            printf("@ 총 %d명이 입력되었습니다.\n", cnt);
      while(1)
                                                             for(i = 0; i < cnt; i++)
22
          if(cnt == 10)
                                                                printf("%s\n", name[i]);
             printf("@ 10명을 모두 입력하였습니다!\n");
25
             break;
                                                    > 이름 : 홍길동
                                                     이름 : 이순신
          printf("> 이름 : ");
28
            anf("%s", temp);
                                                     이름 : 유관순
          if(strcmp(temp, "end") == 0) break;
                                                     이름 : 홍길 🏫
          if(dup_check(name, temp, cnt))
                                                    @ 이름이 이미 등록되었��니다!
                                                    > 이름 : 세종대왕
             printf("@ 이름이 이미 등록되었습니다!\n");
                                                    > 이름 : end
                                                    @ 총 4명이 입력되었습니다.
          strcpy(name[cnt], temp);
                                                    홍길동
          cnt++;
                                                    이 순 신
                                                    유관순
39
                                                    세 종 대 왕
      return cnt;
41 }
```

```
int count_num(int(*pa1)[4], int(*pa2)[4], int(*ps)[2]);
 5 int main(void)
 6 - {
       int ary1[3][4] = {
           {15, 3, 9, 11},
           {23, 7, 10, 12},
           {0, 17, 55, 24}
       int ary2[3][4] = {
           {13, 8, 9, 15},
           {23, 0, 10, 11},
           {29, 17, 43, 3}
       };
       int same_num[12][2];
       int i, cnt;
       cnt = count_num(ary1, ary2, same_num);
       printf("일치하는 번호의 수 : %d\n", cnt);
       printf("배열에서의 위치(행, 열)...\n");
       for (i = 0; i < cnt; i++)
           printf("(%d, %d)\n", same_num[i][0], same_num[i][1]); // 위치 출력
       return 0;
29 }
30 int count_num(int(*pa1)[4], int(*pa2)[4], int(*ps)[2])
       int i, j, cnt = 0;
       for (i = 0; i < 3; i++)
           for (j = 0; j < 4; j++)
               if (pa1[i][j] == pa2[i][j])
                   ps[cnt][0] = i;
                   ps[cnt][1] = j;
                   cnt++;
       return cnt;
46
```

```
일치하는 번호의 수 : 4
배열에서의 위치(행, 열)...
(0, 2)
(1, 0)
(1, 2)
(2, 1)
```

```
3 int main(void)
 4 - {
        char *order[] = {"첫", "두", "세"};
        double ary1[4] = \{1.5, 2.3, 7.7, 4.9\};
        double ary2[4] = \{3.4, 5.2, 4.0, 9.1\};
        double ary3[4] = \{0.3, 7.2, 8.4, 6.6\};
        double *pary[3] = {ary1, ary2, ary3};
        double avg[3];
        double tot, max;
        int i, j, index;
        for (i = 0; i < 3; i++)
            tot = 0;
            for (j = 0; j < 4; j++)
                tot += pary[i][j];
            avg[i] = tot / 4.0;
        max = 0;
        index = 0;
        for (i = 0; i < 3; i++)
            if (avg[i] > max)
                max = avg[i];
                index = i;
        printf("%s 번째 배열 :", order[index]);
        for (i = 0; i < 4; i++)
           printf("%5.1lf", pary[index][i]);
        printf("\n평균: %.1lf", max);
        return 0;
41 }
```

세 번째 배열 : 0.3 7.2 8.4 6.6 평균 : 5.6