

9장. 응용예제 답

응용예제
13

최고의 팀으로 만들라!

bestTeam.c

```
//응용예제13. 최고의 팀 만들기
#define _CRT_SECURE_NO_WARNINGS
#include <stdio.h>
#define MAX 15

int main(void) {
    int N, K, i, k, sum=0, min, max, temp;
    int ourTeam[MAX] = { 0 }, otherTeam[MAX] = { 0 };
    double beforeGrade = 0.0, afterGrade = 0.0;

    //입력: 첫째 줄 => 선수 인원수 N, 트레이드 회수 K
    scanf("%d %d", &N, &K);
    //입력: 둘째 줄 => 우리팀 선수의 성적
    for (i = 0; i < N; i++)
        scanf("%d", &ourTeam[i]);
    //입력: 셋째 줄 => 다른 팀 선수의 성적
    for (i = 0; i < N; i++)
        scanf("%d", &otherTeam[i]);

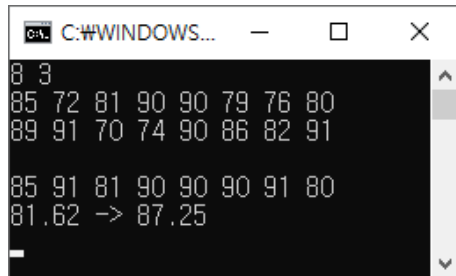
    //우리팀의 현재 성적 평균 구하기
    for (i = 0; i < N; i++)
        sum += ourTeam[i];
    beforeGrade = (double)sum / N;

    for (k = 0; k < K; k++) {
        min = 0; max = 0;
        //우리팀의 최저 성적 찾기
        for (i = 1; i < N; i++)
            if (ourTeam[i] < ourTeam[min]) min = i;
        //다른팀의 최고 성적 찾기
        for (i = 1; i < N; i++)
            if (otherTeam[i] > otherTeam[max]) max = i;
        //트레이드하기
        if (ourTeam[min] >= otherTeam[max]) break;
        else {
            temp = ourTeam[min];
            ourTeam[min] = otherTeam[max];
            otherTeam[max] = temp;
        }
    }

    //트레이드 결과
    sum = 0;
    printf("Wn");
    for (i = 0; i < N; i++) {
```

```
        printf("%d ", ourTeam[i]);  
        sum += ourTeam[i];  
    }  
    afterGrade = (double)sum / N;  
    printf("Wn%.2lf -> %.2lfWn", beforeGrade, afterGrade);  
  
    getch(); return 0;  
}
```

[실행 화면]



```
C:\WINDOWS...  
8 3  
85 72 81 90 90 79 76 80  
89 91 70 74 90 86 82 91  
  
85 91 81 90 90 90 91 80  
81.62 -> 87.25  
_
```

히프정렬을 위해서, [예제 7-6] heap.h 와 heap.c 사용

featureVector.c

```
//응용예제14. 특징벡터 출력하기
#define _CRT_SECURE_NO_WARNINGS
#include <stdio.h>
#include "heap.h"
#define MAX 20

int main(void) {
    int N, M, i, j, h, featureVextorSize, wx, wy;
    int vector[MAX][MAX] = { 0 }, featureVextor[MAX][MAX] = { 0 };
    heapType* heap = createHeap();

    //1. 입력: 첫째 줄 => 데이터크기 N, 윈도우 크기 M
    scanf("%d %d", &N, &M);
    //2. 입력: 데이터 N*N개
    for (i = 0; i < N; i++)
        for (j = 0; j < N; j++)
            scanf("%d", &vector[i][j]);
    //3. 특징벡터 추출하기
    featureVextorSize = N - M + 1;
    for (wx = 0; wx < featureVextorSize; wx++) {
        for (wy = 0; wy < featureVextorSize; wy++) {
            //(1)윈도우 범위에 있는 데이터를 히프에 삽입하기
            for (i = wx; i < (wx+ M); i++)
                for (j = wy; j < (wy+ M); j++)
                    insertHeap(heap, vector[i][j]);
            //(2)히프에서 최대값을 특징벡터에 저장. 나머지 히프원소를 삭제하여 공백 히프
            //만들기
            for (h = 0; h < M * M; h++) {
                if (h == 0) featureVextor[wx][wy] = deleteHeap(heap);

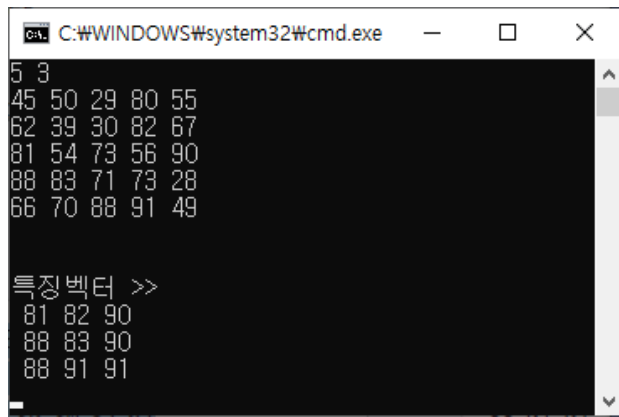
                else deleteHeap(heap);
            }
            //(3) 윈도우를 오른쪽으로 이동
        }
        //(4) 윈도우를 아래쪽으로 이동

    }

    //4. 특징벡터 출력
    printf("N\nM\n특징벡터 >> N\n");
    for (wx = 0; wx < featureVextorSize; wx++) {
        for (wy = 0; wy < featureVextorSize; wy++)
            printf("%3d", featureVextor[wx][wy]);
        printf("\n");
    }

    getch(); return 0;
}
```

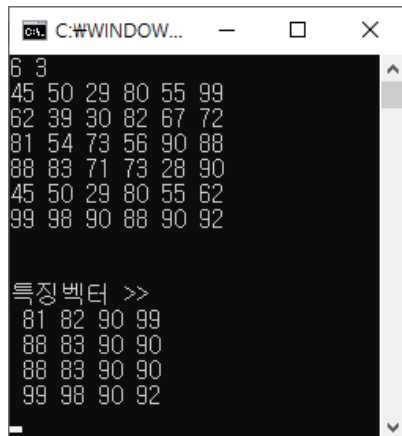
[실행 화면]



```
C:\WINDOWS\system32\cmd.exe
5 3
45 50 29 80 55
62 39 30 82 67
81 54 73 56 90
88 83 71 73 28
66 70 88 91 49

특징벡터 >>
81 82 90
88 83 90
88 91 91
```

[실행 화면] 입력 값이 변경된 경우



```
C:\WINDOW...
6 3
45 50 29 80 55 99
62 39 30 82 67 72
81 54 73 56 90 88
88 83 71 73 28 90
45 50 29 80 55 62
99 98 90 88 90 92

특징벡터 >>
81 82 90 99
88 83 90 90
88 83 90 90
99 98 90 92
```