

```

1. struct m {
    int value;
    int row;
    int column;
    int num
} m1, m2;

```

崔 + 强  
 PB22151743

```

int senpai (int *p, int m, int n) {    // m行 n列
    int (*p)[n] = p, i, j, a = 0;
    m1.value = *p;
    m2.value = *p;
    for (i = 0; i < m; i++) {
        for (j = 0; j < n; j++) {
            if (*(*p + i) + j) > m1.value) {
                m1.value = *(*p + i) + j;
                m1.row = i + 1;
                m1.column = j + 1;
            }

            if (*(*p + i) + j) < m2.value) {
                m2.value = *(*p + i) + j;
                m2.row = i + 1;
                m2.column = j + 1;
            }
        }
    }
    printf("最大值为%d, 最小值为%d\n", m1.value, m2.value);
    m1.num = 0;
    m2.num = 0;
    for (i = 0; i < m; i++) {
        for (j = 0; j < n; j++) {

```

```

if ( *(*(p+i)+j) == m1.value ) {
    m1.num++;
    printf("第%d个最大值在第%d行第%d列,\n",
           m1.num, i+1, j+1);
}

if ( *(*(p+i)+j) == m2.value ) {
    m2.num++;
    printf("第%d个最小值在第%d行第%d列,\n",
           m2.num, i+1, j+1);
}

}

}

return 0

}

```

```

2. int trans (int (*p)[5]) {
    int temp, i, j;
    for (i=0; i<5; i++) {
        for (j=0; j<i; j++) {
            temp = *(*p+i)+j);
            *(*p+i)+j) = *(*p+j)+i);
            *(*p+j)+i) = temp;
        }
    }
    return 0;
}

```

```

int mul (int (*p)[5]) {
    int *p2[5], a[5][5], b[5][5], *p3[5];
    for (i=0; i<5; i++) {
        for (j=0; j<5; j++) {
            a[i][j] = *(*p+i)+j);
        }
    }
    p2 = a;
    p3 = b;
    trans (p2);
    for (i=0; i<5; i++) {
        for (j=0; j<5; j++) {
            for (k=0; k<5; k++) {
                *(*p3+i)+j) = (*(*p+i)+k)) * (*(*p2+k)+j));
            }
        }
    }
}

```

```
for(i=0; i<5; i++){  
    for(j=0; j<5; j++){  
        printf("%d\t", *(*(p3+i)+j));  
    }  
    printf("\n");  
}  
return 0;  
}
```

```

1. void sort(struct stu *p_array, int score_num, int num){
    int i, j;
    struct stu temp;
    switch(score_num){
        case 1:
            for(i=0; i<num-1; i++){
                for(j=0; j<num-1; j++){
                    if((*(p_array+j)).score_1 > (*(p_array+j+1)).score_1){
                        temp = (*(p_array+j)).score_1;
                        (*(p_array+j)).score_1 = (*(p_array+j+1)).score_1;
                        (*(p_array+j+1)).score_1 = temp;
                    }
                }
            }
            break;
        case 2:
            for(i=0; i<num-1; i++){
                for(j=0; j<num-1; j++){
                    if((*(p_array+j)).score_2 > (*(p_array+j+1)).score_2){
                        temp = (*(p_array+j)).score_2;
                        (*(p_array+j)).score_2 = (*(p_array+j+1)).score_2;
                        (*(p_array+j+1)).score_2 = temp;
                    }
                }
            }
            break;
    }
}

```

case 3:

```
for(i=0; i<num-1; i++){  
    for(j=0; j<num-1; j++){  
        if((*(p_array+j)).score_3 > (*(p_array+j+1)).score_3){  
            temp = (*(p_array+j)).score_3;  
            (*(p_array+j)).score_3 = (*(p_array+j+1)).score_3;  
            (*(p_array+j+1)).score_3 = temp;  
        }  
    }  
}  
break;  
}
```

```

2. void sort(int **p, int n){
    int temp
    for(i=0; i<n-1; i++){
        for(j=0; j<n-1; j++){
            if (**(p+j) < *(p+j+1)) {
                temp = *(p+j);
                *(p+j) = *(p+j+1);
                *(p+j+1) = temp;
            }
        }
    }
}

```