

```
//第一题
#include <stdio.h>
int countAboveAverage(int scores[], int count) {
    int sum = 0;
    float average;
    int aboveCount = 0;

    // 计算总分
    for (int i = 0; i < count; i++) {
        sum += scores[i];
    }

    // 计算平均分
    average = (float)sum / count;

    // 统计高于平均分的人数
    for (int i = 0; i < count; i++) {
        if (scores[i] > average) {
            aboveCount++;
        }
    }

    return aboveCount;
}

int main() {
    int scores[40];
    int count = 0;
    int score;

    printf("请输入学生成绩（输入负值结束）: \n");

    // 输入成绩
    while (count < 40) {
        scanf("%d", &score);
        if (score < 0) {
            break;
        }
        scores[count] = score;
        count++;
    }

    if (count == 0) {
        printf("没有输入有效成绩! \n");
    }
}
```

```
        return 0;
    }

    int result = countAboveAverage(scores, count);
    printf("成绩高于平均分的学生人数: %d\n", result);

    return 0;
}

//第二题
//1、c[ROW][ROW]

//2、0

//3、c[i][j] + a[i][k] * b[k][j] 或 c[i][j] += a[i][k] * b[k][j]

//4、printf("\n");

//5、&a[i][j]

//6、&b[i][j]

//7、a, b, c

//第三题
#include <stdio.h>
#define N 10

void Swap(int *x, int *y) {
    int temp;
    temp = *x;
    *x = *y;
    *y = temp;
}

void Transpose(int a[][N], int n) {
    int i, j;
    for (i = 0; i < n; i++) {
        for (j = i + 1; j < n; j++) {
            // 交换 a[i][j] 和 a[j][i]
            Swap(&a[i][j], &a[j][i]);
        }
    }
}
```

```
void PrintMatrix(int a[][N], int n) {
    int i, j;
    for (i = 0; i < n; i++) {
        for (j = 0; j < n; j++) {
            printf("%4d", a[i][j]);
        }
        printf("\n");
    }
}

int main() {
    int n;
    int matrix[N][N];
    int i, j;

    printf("请输入矩阵的阶数 n (n≤10) : ");
    scanf("%d", &n);

    if (n > 10 || n <= 0) {
        printf("输入的阶数不合法! \n");
        return 1;
    }

    printf("请输入%d×%d 矩阵: \n", n, n);
    for (i = 0; i < n; i++) {
        for (j = 0; j < n; j++) {
            scanf("%d", &matrix[i][j]);
        }
    }

    printf("原始矩阵: \n");
    PrintMatrix(matrix, n);

    Transpose(matrix, n);

    printf("转置矩阵: \n");
    PrintMatrix(matrix, n);

    return 0;
}
```