answers05

2.

```c
#include <stdio.h>
int main()
{
    int data[100];
    int n = 0;
    scanf("%d", &n);
    for (int i=0; i<n; i++)
        scanf("%d", &data[i]);

    int smallest = data[0], secondSmallest = data[1];
    if (data[0] > data[1]) {
        smallest = data[1];
        secondSmallest = data[0];
    }
    for (int i=2; i<n; i++) {
        if (data[i] <= smallest) {
            secondSmallest = smallest;
            smallest = data[i];
        }
        else if (data[i] < secondSmallest)
            secondSmallest = data[i];
    }
    printf("%d %d\n", smallest, secondSmallest);
}
```

3.

```c
#include <stdio.h>
#include <math.h>
int main()
{
    int data[100];
    int n = 0, sum = 0;
    scanf("%d", &n);
    for (int i=0; i<n; i++) {
        scanf("%d", &data[i]);
        sum += data[i];
    }
    double average = (double)sum / n;
    double sumSq = 0.0;
    for (int i=0; i<n; i++)
        sumSq += (data[i]-average)*(data[i]-average);
    printf("%lf %lf\n", average, sqrt(sumSq/n));
}
```

5.

```c
#include <stdio.h>
int main()
{
    int daysInMonth[] = { 0, 31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31 };
    int y, m, d, by, bm, bd;

    scanf("%d %d %d", &by, &bm, &bd);
    scanf("%d %d %d", &y, &m, &d);

    int daysThisYearToday = 0;
    for (int i=1; i<m; i++)
        daysThisYearToday += daysInMonth[i];
```

```
    daysThisYearToday += d;

    int daysThisYearBirth = 0;
    for (int i=1; i<bm; i++)
        daysThisYearBirth += daysInMonth[i];
    daysThisYearBirth += bd;

    int daysPassed = (y-by)*365 + daysThisYearToday - daysThisYearBirth;
    printf("%d\n", daysPassed);
}
```

6.

```
#include <stdio.h>
int main()
{
    int data[100];
    int n = 0, sum = 0, k;
    scanf("%d", &n);
    for (int i=0; i<n; i++)
        scanf("%d", &data[i]);
    scanf("%d", &k);
    int minDiff = (k-data[0] > 0 ? k-data[0] : data[0] - k);
    int minIndex = 0;
    for (int i=1; i<n; i++) {
        int diff = (k-data[i] > 0 ? k-data[i] : data[i] - k);
        if (diff < minDiff) {
            minDiff = diff;
            minIndex = i;
        }
    }
    printf("%d\n", data[minIndex]);
}
```

7.

```
#include <stdio.h>
int main()
{
    int data[100];
    int n = 0;
    FILE *fp = fopen("input7.txt", "r");
    while(!feof(fp))
        fscanf(fp, "%d", &data[n++]);
    fclose(fp);

    int last = data[n-1];
    for (int i=n-1; i>0; i--)
        data[i] = data[i-1];
    data[0] = last;

    for (int i=0; i<n; i++)
        printf("%d ", data[i]);
}
```

8.

```
#include <stdio.h>
int main()
{
    int data[100];
    int n = 0, k;
    FILE *fp = fopen("input8.txt", "r");
```

```
    while(fscanf(fp, "%d", &k) != EOF)
        data[n++] = k;
    fclose(fp);

    for (int i=0; i<n/2; i++) {
        int tmp = data[i];
        data[i] = data[n-i-1];
        data[n-i-1] = tmp;
    }

    for (int i=0; i<n; i++)
        printf("%d ", data[i]);
}
```

9.

```
#include <stdio.h>
int main()
{
    int data[100];
    int n = 0, k;
    FILE *fp = fopen("input9.txt", "r");
    while(fscanf(fp, "%d", &k) != EOF)
        data[n++] = k;
    fclose(fp);

    int t = 0, maxSoFar = data[0];
    for (int i=1; i<n; i++) {
        if (data[i] >= maxSoFar) {
            maxSoFar = data[i];
            t++;
            data[t] = data[i];
        }
    }
    printf("%d ", t+1);
    for (int i=0; i<=t; i++)
        printf("%d ", data[i]);
}
```

11.

```
#include <stdio.h>
int main() {
    int data1[100], data2[100], data3[200];
    int n = 0;
    FILE *fp1 = fopen("input11_1.txt", "r");
    FILE *fp2 = fopen("input11_2.txt", "r");
    while(!feof(fp1)) {
        fscanf(fp1, "%d", &data1[n]);
        fscanf(fp2, "%d", &data2[n++]);
    }
    fclose(fp1);
    fclose(fp2);

    for (int i=0; i<n; i++) {
        data3[2*i] = data1[i];
        data3[2*i+1] = data2[i];
    }
    for (int i=0; i<2*n; i++)
        printf("%d\n", data3[i]);
}
```

12.

```c
#include <stdio.h>
#include <math.h>
int main()
{
    int x[100], y[100];
    int n = 0;
    int sumX = 0, sumY = 0;
    FILE *fp = fopen("input12.txt", "r");
    while(!feof(fp)) {
        fscanf(fp, "%d %d", &x[n], &y[n]);
        sumX += x[n];
        sumY += y[n];
        n++;
    }
    fclose(fp);

    double centerX = (double)sumX/n;
    double centerY = (double)sumY/n;
    double sumDist = 0.0;

    for (int i=0; i<n; i++)
        sumDist += sqrt( (x[i]-centerX)*(x[i]-centerX)
                         + (y[i]-centerY)*(y[i]-centerY) );
    printf("%lf %lf %lf", centerX, centerY, sumDist/n);
}
```