

高性能并行计算第 1 次作业

姓名: 张子栋 学号: 2020317210101

代码地址: `/home/2020317210101/work1`

`https://github.com/Bluuur/MarkdownNotes/tree/main/高性能并行计算/Code1`

实验结果

1. 计算 N 维数据的最大值, 最小值, 平均值, 标准差

代码

```
1 //
2 // Created by ZidongZh on 2022/9/23.
3 //
4
5 #include "stdio.h"
6 #include "math.h"
7
8 double getMax(double array[]) {
9     double max = array[0];
10    for (int i = 0; i < ((int) sizeof(array) /
11        sizeof(*array)); ++i) {
12        if (array[i] >= max) {
13            max = array[i];
14        }
15    }
16    return max;
17 }
18
19 double getMin(double array[]) {
20     double min = array[0];
21     for (int i = 0; i < ((int) sizeof(array) /
22         sizeof(*array)); ++i) {
23         if (array[i] <= min) {
24             min = array[i];
25         }
26     }
27     return min;
28 }
29
30 double getMean(double array[]) {
```

```

29     int len = (int) sizeof(array) / sizeof(*array);
30     double sum = 0.0;
31     double mean = 0.0;
32     for (int i = 0; i < len; ++i) {
33         sum += array[i];
34     }
35     mean = sum / len;
36     return mean;
37 }
38
39 double getSD(double array[]) {
40     int len = (int) sizeof(array) / sizeof(*array);
41     double mean = getMean(array);
42     double SS = 0.0;
43     for (int i = 0; i < len; ++i) {
44         SS += pow(array[i] - mean, 2);
45     }
46     double SD = sqrt(SS / (len - 1));
47     return SD;
48 }
49
50 int main() {
51
52     // Get the length of data
53     int length;
54
55     printf("Enter the size of data:");
56     scanf("%d", &length);
57
58     // init array, get data
59     double array[length];
60
61     for (int i = 0; i < length; ++i) {
62         printf("Enter data (%d of %d):", i + 1, length);
63         scanf("%lf", &array[i]);
64     }
65
66     // Display data
67     printf("Entered data:");
68
69     for (int i = 0; i < length; ++i) {
70         printf("%lf ", array[i]);
71     }
72
73     // Output

```

```

74     printf("\nMax of data is: %lf\n", getMax(array));
75     printf("Min of data is: %lf\n", getMin(array));
76     printf("Mean of data is: %lf\n", getMean(array));
77     printf("SD of data is: %lf\n", getSD(array));
78 }

```

结果

```

1  Enter the size of data:4
2  Enter data (1 of 4):12.6
3  Enter data (2 of 4):15.4
4  Enter data (3 of 4):19.8
5  Enter data (4 of 4):11.5
6  Entered data:12.600000 15.400000 19.800000 11.500000
7  Max of data is: 12.600000
8  Min of data is: 12.600000
9  Mean of data is: 12.600000
10 SD of data is: 9.970886

```

2. 计算 N 维向量点乘

代码:

```

1  //
2  // Created by zidongzh on 2022/9/23.
3  //
4
5  #include <stdio.h>
6
7  int main() {
8      int dimension;
9
10     // Get the dimension of array
11     printf("Enter the dimension of the two vectors:");
12     scanf("%d", &dimension);
13
14     // Initialize array
15     double array1[dimension];
16     double array2[dimension];
17
18     // Get data
19     for (int i = 0; i < dimension; ++i) {
20         printf("Enter data (vector1, %d of %d):", i + 1,
21             dimension);
22         scanf("%lf", &array1[i]);

```

```

22     }
23     printf("-----\n");
24     for (int i = 0; i < dimension; ++i) {
25         printf("Enter data (vector2, %d of %d):", i + 1,
dimension);
26         scanf("%lf", &array2[i]);
27     }
28
29     // Display
30     printf("\nvector 1:\n");
31     for (int i = 0; i < dimension; ++i) {
32         printf("%lf \n", array1[i]);
33     }
34     printf("\nVector 2:\n");
35     for (int i = 0; i < dimension; ++i) {
36         printf("%lf\n", array2[i]);
37     }
38
39     // Compute & Output
40     double result = 0;
41     for (int i = 0; i < dimension; ++i) {
42         result += (array1[i] * array2[i]);
43     }
44
45     printf("\nDot product of the two vectors is %lf", result);
46
47     return 0;
48 }
49

```

结果

```

1  Enter the dimension of the two vectors:4
2  Enter data (vector1, 1 of 4):1
3  Enter data (vector1, 2 of 4):2
4  Enter data (vector1, 3 of 4):3
5  Enter data (vector1, 4 of 4):4
6  -----
7  Enter data (vector2, 1 of 4):5
8  Enter data (vector2, 2 of 4):6
9  Enter data (vector2, 3 of 4):7
10 Enter data (vector2, 4 of 4):8
11
12 Vector 1:
13 1.000000

```

```
14 2.000000
15 3.000000
16 4.000000
17
18 Vector 2:
19 5.000000
20 6.000000
21 7.000000
22 8.000000
23
24 Dot product of the two vectors is 70.000000
```

3. 计算 N 维矩阵点乘

```
1 //
2 // Created by zidongZh on 2022/9/23.
3 //
4
5 #include <stdio.h>
6
7 int main() {
8     int dimension;
9
10    // Get the dimension of array
11    printf("Enter the dimension of the matrix:");
12    scanf("%d", &dimension);
13
14    // Initialize array
15    double matrix1[dimension][dimension];
16    double matrix2[dimension][dimension];
17
18    //
19    for (int i = 0; i < dimension; ++i) {
20        for (int j = 0; j < dimension; ++j) {
21            printf("Enter the number of matrix1 at row %d,
22            column %d:", i + 1, j + 1);
23            scanf("%lf", &matrix1[i][j]);
24        }
25        printf("-----\n");
26        for (int i = 0; i < dimension; ++i) {
27            for (int j = 0; j < dimension; ++j) {
28                printf("Enter the number of matrix2 at row %d,
29                column %d:", i + 1, j + 1);
30                scanf("%lf", &matrix2[i][j]);
```

```

30     }
31 }
32
33 printf("\nMatrix 1:\n");
34 for (int i = 0; i < dimension; ++i) {
35     for (int j = 0; j < dimension; ++j) {
36         printf("%1f ", matrix1[i][j]);
37     }
38     printf("\n");
39 }
40 printf("\nMatrix 2:\n");
41 for (int i = 0; i < dimension; ++i) {
42     for (int j = 0; j < dimension; ++j) {
43         printf("%1f ", matrix2[i][j]);
44     }
45     printf("\n");
46 }
47
48 // Compute
49 double result = 0;
50 for (int i = 0; i < dimension; ++i) {
51     for (int j = 0; j < dimension; ++j) {
52         result += (matrix1[i][j] * matrix2[i][j]);
53     }
54 }
55
56 printf("\nDot product of the two vectors is %1f", result);
57 return 0;
58 }

```

结果

```

1 Enter the dimension of the matrix:2
2 Enter the number of matrix1 at row 1, column 1:1
3 Enter the number of matrix1 at row 1, column 2:2
4 Enter the number of matrix1 at row 2, column 1:3
5 Enter the number of matrix1 at row 2, column 2:4
6 -----
7 Enter the number of matrix2 at row 1, column 1:5
8 Enter the number of matrix2 at row 1, column 2:6
9 Enter the number of matrix2 at row 2, column 1:7
10 Enter the number of matrix2 at row 2, column 2:8
11
12 Matrix 1:
13 1.000000 2.000000

```

```
14 3.000000 4.000000
15
16 Matrix 2:
17 5.000000 6.000000
18 7.000000 8.000000
19
20 Dot product of the two vectors is 70.000000
21 Process finished with exit code 0
22
```

4. 大量随机数冒泡排序

```
1 //
2 // Created by zidongzh on 2022/9/23.
3 //
4
5 #include<stdio.h>
6 #include<stdlib.h>
7
8 void BubbleSort(int array[], int length) {
9     int i, j, temp;
10    for (i = 0; i < length - 1; i++) {
11        for (j = 0; j < length - i - 1; j++) {
12            if (array[j] > array[j + 1]) {
13                temp = array[j + 1];
14                array[j + 1] = array[j];
15                array[j] = temp;
16            }
17        }
18    }
19 }
20
21 int main() {
22     // Get the length of data
23     int length;
24
25     printf("Enter the size of data:");
26     scanf("%d", &length);
27
28     int array[length];
29
30     for (int i = 0; i < length; i++) {
31         array[i] = rand();
32     }
33     BubbleSort(array, length);
```

```
34
35     for (int j = 0; j < length; j++) {
36         printf("%d ", array[j]);
37     }
38
39     return 0;
40 }
```

结果

```
1 Enter the size of data:10
2 41 6334 11478 15724 18467 19169 24464 26500 26962 29358
```