

Mesh09 Airfoil Application

19335015 陈恩婷

1. 项目目的

根据课件和程序中给出的函数原型和提示，实现 Program1.cpp 中的 run 函数并修正

代码中的错误。相关课件截图如下：

- Complete **run()** function in Program1 *领域应用程序*
- Modify other related functions

主要需要修改的地方如下：

```
int Program1::run()
{
    ....
    for (int iter = 1; iter <= niter; iter++) {
        for (int i = 0; i < n; i++) {
            save((double*)s.dat_list[2]->data, (double*)s.dat_list[3]->data);
        }
        for (int k = 0; k < 2; k++) {
            for (int i = 0; i < n; i++) { ///////////////
                area((double*)s.dat_list[1]->data+xxxxx, (double*)s.dat_list[1]->data+xxxxx,
                    (double*)s.dat_list[1]->data, (double*)s.dat_list[1]->data,
                    (double*)s.dat_list[2]->data, (double*)s.dat_list[4]->data);
            }
            for (int i = 0; i < m; i++) { ///////////////
                flux((double*)s.dat_list[1]->data, (double*)s.dat_list[1]->data + 1,
                    (double*)s.dat_list[2]->data, (double*)s.dat_list[2]->data + 1,
                    (double*)s.dat_list[4]->data, (double*)s.dat_list[4]->data + 1,
                    (double*)s.dat_list[5]->data, (double*)s.dat_list[5]->data + 1);
            }
            for (int i = 0; i < bm; i++) { ///////////////
                bcond((double*)s.dat_list[1]->data, (double*)s.dat_list[1]->data + 1,
                    (double*)s.dat_list[2]->data, (double*)s.dat_list[4]->data,
                    (double*)s.dat_list[5]->data, (int*)s.dat_list[0]->data);
            }
            rms = 0.0;
            for (int i = 0; i < n; i++) {
                update((double*)s.dat_list[3]->data, (double*)s.dat_list[2]->data,
                    (double*)s.dat_list[5]->data, (double*)s.dat_list[4]->data, &rms);
            }
            rms = sqrt(rms / (double)n);
        }
        ....
    }
}
```

对于各Kernel函数调用

*调用参数需要修改
参考 airfoil_seq.cpp 代码。*

工程中原有的代码可以根据实际情况进行改动，尤其是 CHECK_DOUBLES_EQUAL 部分。

2. 为实现目的存在的各种技术问题

熟练掌握指针作为函数参数的写法

参考课件中的 airfoil_seq.cpp 程序，理解清楚需要实现的函数的计算逻辑，尤其是传入

各个核函数的指针是如何计算得到的。

3. 用什么算法、数据结构、语言机制解决这些问题

数据结构：主要是 mesh 中的 elements, maps 和 data 等结构体，重点是结构体中的
是 double *和 int *等指针代表的数组成员。

语言机制：将指针作为参数传递给函数

要注意计算方法,比如传递 a[i]处的指针需要写做 a+i.参考 airfoil_seq 中的数组写法,

在 program1.cpp 中转换成指针写法即可。

4. 对应的程序框架和实现代码

Mesh 部分和 Program 部分的代码和之前的实验基本一致，这里就不赘述了。

Program1 类的声明与框架

```
class Program1
{
    double gam, gm1, cfl, eps, mach, alpha, qinf[4]; //7 constants
    Mesh & s;
    //4 elements: nodes,edges,bedges, cells
    //5 maps: cell2node,edge2node,edge2cell,bedge2node, bedge2cell
    //6 dat: bound(bedge,int), x(nodes), q(4 on cells), qold(4 on cells),
    adt(cell), res(4 on cell)

public:
    void initflow();
    //Kernel functions
    void save(const double* q, double* qold);
    void area(const double* x1, const double* x2, const double* x3,const
double* x4, const double* q, double* adt);
    void flux(const double* x1, const double* x2, const double* q1,
const double* q2, const double* adt1, const double* adt2, double* res1,
double* res2);
    void bcond(const double* x1, const double* x2, const double* q1,
const double* adt1, double* res1, const int* bound);// Apply boundary
conditions
    void update(const double* qold, double* q, double* res, const double* adt,
double* rms);

    //Program
    int run(int step);
```

```

public:
    Program1(Mesh &ms);
    ~Program1();
};

```

以上是 Program1 类的声明，主要需要自己实现的函数主要就是 run。关键代码

如下：

```

int Program1::run(int step)
{
    initflow();

    int niter = 100;
    int n = s.element_list[3]->size; //ncells
    int m = s.element_list[1]->size; //nedges
    int bm = s.element_list[2]->size; //nbedges
    double rms=0.0;
    for (int iter = 1; iter <= niter; iter++) {
        for (int i = 0; i < n; i++) {
            save((double*)s.dat_list[2]->data + 4*i,
(double*)s.dat_list[3]->data + 4*i );
        }
        for (int k = 0; k < 2; k++) {
            for (int i = 0; i < n; i++) { ////////////
                area((double*)s.dat_list[1]->data+3*(s.map_list[4]->map[4*i]),
(double*)s.dat_list[1]->data + 3*(s.map_list[4]->map[4*i + 1]),
(double*)s.dat_list[1]->data+3*(s.map_list[4]->map[4 * i + 2]),
                (double*)s.dat_list[1]->data+3*(s.map_list[4]->map[4*i+3]),
(double*)s.dat_list[2]->data + 4*i, (double*)s.dat_list[4]->data + i);
            }
            for (int i = 0; i < m; i++) { ////////////
                flux((double*)s.dat_list[1]->data+3*(s.map_list[0]->map[2*i]),
(double*)s.dat_list[1]->data + 3*(s.map_list[0]->map[2*i+1]),
(double*)s.dat_list[2]->data+4*(s.map_list[1]->map[2*i]),
                (double*)s.dat_list[2]->data+4*(s.map_list[1]->map[2*i+1]),
(double*)s.dat_list[4]->data+s.map_list[1]->map[2*i],
(double*)s.dat_list[4]->data + s.map_list[1]->map[2 * i + 1],
                (double*)s.dat_list[5]->data + 4*(s.map_list[1]->map[2 *
i]), (double*)s.dat_list[5]->data+4*(s.map_list[1]->map[2 * i + 1]));
            }
            for (int i = 0; i < bm; i++) { ////////////

```

```

        bcond((double*)s.dat_list[1]->data + 3 * (s.map_list[2]->map[2 *
i]), (double*)s.dat_list[1]->data + 3 * (s.map_list[2]->map[2 * i + 1]),
(double*)s.dat_list[2]->data + 4*(s.map_list[3]->map[i]),
        (double*)s.dat_list[4]->data + s.map_list[3]->map[i],
(double*)s.dat_list[5]->data + 4 * (s.map_list[3]->map[i]),
(int*)s.dat_list[0]->data + i);
    }
    rms = 0.0;

    for (int i = 0; i < n; i++) {
        update((double*)s.dat_list[3]->data + 4*i,
(double*)s.dat_list[2]->data + 4*i, (double*)s.dat_list[5]->data + 4*i,
        (double*)s.dat_list[4]->data + i, &rms);
    }
    double* p9 = (double*)s.dat_list[2]->data;
}
rms = sqrt(rms / (double)n);
}
return 0;
}

```

5. 实验结果和结论

先运行参考代码 airfoil_seq.cpp, 结果如下:

```

Microsoft Visual Studio 调试控制台
□: 25642
□: 25042
Writing OutputSimulation to ASCII file: out90.vtk
100 5.25536e-04
    save: 0.192920
    area: 2.143996
    flux_res: 5.450028
    perem: 0.020237
    update: 0.794044
180000 24962
□: 25563
□: 25563
□: 25564
□: 24963
180000 25041
□: 25042
□: 25643
□: 25642
□: 25042
Writing OutputSimulation to ASCII file: out100.vtk
1
-1.03946e-16
1
Max total runtime =
32.228359
C:\Users\豹豹\OneDrive - 中山大学\大三上\程序设计\Mesh09\Airfoil\x64\Debug\Airfoil.exe (进程 11496)已退出, 代码为 0。
要在调试停止时自动关闭控制台, 请启用“工具”->“选项”->“调试”->“调试停止时自动关闭控制台”。
按任意键关闭此窗口。 . . .

```

用参考程序相关数组中的内容修改 program1.cpp 中的 CHECK_DOUBLES_EQUAL 部分, 再

运行程序, 发现结果和参考程序一致:

```
Microsoft Visual Studio 调试控制台

writing in grid
20 1.41026e-03
writing in grid
30 1.07837e-03
writing in grid
40 9.13047e-04
writing in grid
50 7.99790e-04
writing in grid
60 7.18167e-04
writing in grid
70 6.55091e-04
writing in grid
80 6.04079e-04
writing in grid
90 5.61625e-04
writing in grid
100 5.25536e-04
writing in grid
1 tests run
There were no test failures
Your grade is 800
请按任意键继续. . .

C:\Users\豹豹\OneDrive - 中山大学\大三上\程序设计\Mesh09\Mesh09.AirfoilApp\Debug\zgMeshApp.exe (进程 32072)已退出，代码为 0。
要在调试停止时自动关闭控制台，请启用“工具”->“选项”->“调试”->“调试停止时自动关闭控制台”。
按任意键关闭此窗口. . .
```