

**C++程序设计实验报告**



实验题目 输入输出流

学 号 201806061108

姓 名 胡皓睿

班 级 计实1801

提交日期 2019.5.22

1. 实验目的：
2. 掌握输入输出流的使用
3. 实现文件的读写。

1. 实验内容：
2. 从键盘输入一批数值，要求保留三位小数，在输出时上下行小数点对齐
3. 编写程序，在显示屏上显示一个由字母B形成的三角形
4. 建立两个磁盘文件f1.dat和f2.dat，编写程序实现以下操作
   1. 从键盘读入20个整数，分别存放在两个磁盘文件中
   2. 从f1.dat中读入10个整数，然后存放到f2.dat原有数据的后面
   3. 从f2.dat中读入20个整数，将它们按从小到大的顺序存放到f2.dat（不保留原来的数据）
5. 在例13.15的基础上，修改程序，将存放在c数组中的元素读入并显示出来

三、实验过程

1. 从键盘输入一批数值，要求保留三位小数，在输出时上下行小数点对齐
   1. 实验代码

#include <iostream>

#include <iomanip>

using namespace std;

int main()

{

double num[1000];

int k = 0;

while (cin >> num[k])

{

k++;

}

for (int x = 0; x < k; x++)

{

cout.width(15);

cout.precision(3);

cout.setf(ios::fixed);

cout << num[x] << endl;

}

return 0;

}

* 1. 实验截图



* 1. 实验结果

使用了cout对象的成员函数控制了其的输出

1. 编写程序，在显示屏上显示一个由字母B形成的三角形
   1. 实验代码

#include <iostream>

using namespace std;

int main()

{

for (int x = 0; x < 8; x++)

{

for (int y = 0; y < 7 - x; y++)

{

cout << ' ';

}

for (int y = 0; y < 2 \* x + 1; y++)

{

cout << 'B';

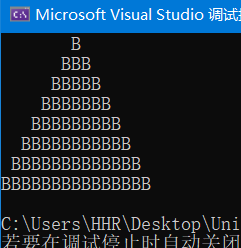
}

cout << endl;

}

}

* 1. 实验截图



* 1. 实验结果

水题，不解释

1. 建立两个磁盘文件f1.dat和f2.dat，编写程序实现以下操作
   1. 从键盘读入20个整数，分别存放在两个磁盘文件中
   2. 从f1.dat中读入10个整数，然后存放到f2.dat原有数据的后面
   3. 从f2.dat中读入20个整数，将它们按从小到大的顺序存放到f2.dat（不保留原来的数据）
      1. 实验代码

#include <iostream>

#include <fstream>

#include <algorithm>

using namespace std;

int main()

{

ofstream outfile;

ifstream infile;

outfile.open("f1.dat", ios::out);

outfile.close();

outfile.open("f2.dat", ios::out);

outfile.close();

int num[20];

for (int x = 0; x < 20; x++)

{

cin >> num[x];

}

outfile.open("f1.dat", ios::out);

for (int x = 0; x < 10; x++)

{

outfile << num[x] << endl;

}

outfile.close();

outfile.open("f2.dat", ios::out);

for (int x = 10; x < 20; x++)

{

outfile << num[x] << endl;

}

outfile.close();

infile.open("f1.dat", ios::in);

for (int x = 0; x < 10; x++)

{

infile >> num[x];

}

infile.close();

outfile.open("f2.dat", ios::app);

for (int x = 0; x < 10; x++)

{

outfile << num[x] << endl;

}

outfile.close();

infile.open("f2.dat", ios::in);

for (int x = 0; x < 20; x++)

{

infile >> num[x];

}

infile.close();

sort(num, num + 20);

outfile.open("f2.dat", ios::out);

for (int x = 0; x < 20; x++)

{

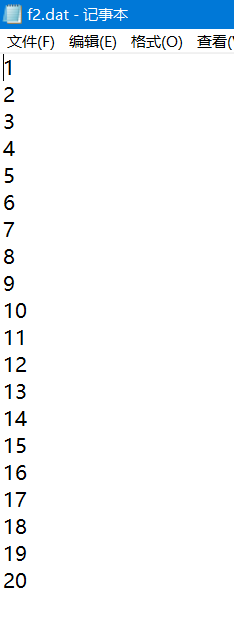
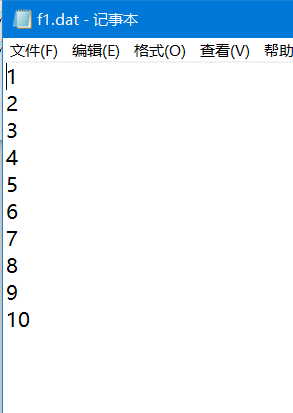
outfile << num[x] << endl;

}

outfile.close();

}

* + 1. 实验截图



* + 1. 实验结果

掌握了使用ofstream和ifstream进行文件输入输出的操作

1. 在例13.15的基础上，修改程序，将存放在c数组中的元素读入并显示出来
   1. 实验代码

#include <strstream>

#include <iostream>

using namespace std;

int main()

{

char c[50] = "12 34 65 -23 -32 33 61 99 321 32";

int a[10], i, j, t;

cout << "array c:" << c << endl;

istrstream strin(c, sizeof(c));

for (i = 0; i < 10; i++)

strin >> a[i];

cout << "array a:";

for (i = 0; i < 10; i++)

cout << a[i] << " ";

cout << endl;

for (i = 0; i < 9; i++)

for (j = 0; j < 9 - i; j++)

if (a[j] > a[j + 1])

{

t = a[j];

a[j] = a[j + 1];

a[j + 1] = t;

}

ostrstream strout(c, sizeof(c));

for (i = 0; i < 10; i++)

strout << a[i] << " ";

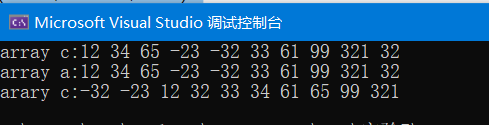
strout << ends;

cout << "arary c:" << c << endl;

return 0;

}

* 1. 实验截图



* 1. 实验结果

皆大欢喜！

【毕业要求1】能够将数学、自然科学、工程基础和专业知识用于解决复杂软件工程问题。

支撑指标点1.3 能够将工程基础和专业知识用于求解软件领域复杂工程问题

【毕业要求3】能够设计针对复杂工程问题的解决方案，设计满足特定需求的系统、单元（部件）或工艺流程，并能够在设计环节中体现创新意识，考虑社会、健康、安全、法律、文化以及环境等因素

支撑指标点3.1：掌握解决复杂工程问题的程序设计语言基础。

【毕业要求5】使用现代工具：能够针对复杂工程问题，开发、选择与使用恰当的技术、资源、现代工程工具和信息技术工具，包括对复杂工程问题的预测与模拟，并能够理解其局限性

支撑指标点5.1：了解现代工程工具和信息技术工具的使用方法,了解实际计算机工程系统中各种现代工具的使用现状。