思考与练习：

（1）本例并没有把重载的运算符>>和<<声明为 DoubleArray 的友元，但却能访问到 DoubleArray 对象中的数组元素，这是怎么做到的？

（2）在>>的重载函数中，第二参数声明为 DoubleArray &a；能否改为 const DoubleArray &a？能否改为 DoubleArray a？若能，说明它们之间的区别；若不能，请说明原因。

（3）在<<的重载函数中，第二参数声明为 const DoubleArray &a；能否改为 DoubleArray a？能否改为 DoubleArray &a？若能，说明它们之间的区别；若不能，请说明原因。

## 小　结

本章主要学习如何使用流进行输入/输出。首先介绍了流的基本概念，其次讲述了无格式和有格式的输入/输出方法和注意事项，然后介绍了流在文件输入/输出中的应用和字符串流，最后介绍了插入运算符<<和提取运算符>>在自定义类型中的重载。

## 习　题

### 一、选择题

1. I/O 流中的预定义流对象包括（　　）。

A. cin、cout、cerr、clog　　B. cin、cout、cerr、ios

C. cin、cout、ios、clog　　D. cin、ios、cerr、clog

2. I/O 流中重载了运算符 <<，它是一个（　　）。

A. 用于输出操作的成员函数　　B. 用于输入操作的成员函数

C. 用于输入操作的非成员函数　　D. 用于输出操作的非成员函数

3. 对于语句 cout<<endl<<x; 中的各个组成部分，下列解释中错误的是（　　）。

A. “cout”是一个输出流对象　　B. “endl”的作用是回车换行

C. “<<”称作提取运算符　　D. “x”是一个变量

4. 语句 ofstream f("DATA.DAT",ios::app | ios::binary);的功能是建立流对象 f，并试图打开文件 DATA.DAT 并与之连接，而且（　　）。

A. 若文件存在，将文件定位于文件首；若文件不存在，建立一个新文件

B. 若文件存在，将其截为空文件；若文件不存在，打开失败

C. 若文件存在，将文件定位于文件尾；若文件不存在，建立一个新文件

D. 若文件存在，打开失败；若文件不存在，建立一个新文件

5. 在进行了任何 I/O 流的操作后，都可以用 I/O 流的有关成员函数检测流的状态；下列函数名中，只能用于检测输入操作的函数的函数名是（　　）。

A. fail　　B. eof　　C. bad　　D. good

6. 执行语句序列

```
ofstream outfile("DATA.DAT");
```

```
if(<条件>) cout<<"OK"; else cout<<"FAIL";
```

后，如果文件打开成功，显示“OK”，否则显示“FAIL”。由此可知，上面 if 语句的<条件>应当是（　　）。

A. outfile.fail() 或 outfile　　B. outfile.good() 或 !outfile

C. outfile.good() 或 outfile　　D. outfile.fail() 或 !outfile

7. 在 I/O 流类中既可以用于文件输入，又可以用于文件输出的流类是（　　）。

A. fstream　　B. ifstream　　C. ofstream　　D. iostream

8. 有以下程序：

```
#include <iostream>
#include <iomanip>
using namespace std;
int main(){
    cout << setfill('#') << setw(4) << "OK" << 123 << endl;
    return 0;
}
```

执行后的输出结果是（　　）。

A. ##OK123　　B. ##OK#123　　C. OK##123　　D. OK##123#

9. 已知一程序运行后执行的第 1 个输入/输出语句是

```
cout<<setfill('*')<<setw(8)<<x;
```

关于此语句的输出效果，下列表述中正确的是（　　）。

A. 如果数据的实际宽度大于 8，数据前部的多余部分被截去

B. 如果数据的实际宽度大于 8，数据后部的多余部分被截去

C. 如果数据的实际宽度小于 8，左边显示若干个 * ，右边显示数据

D. 如果数据的实际宽度小于 8，右边显示若干个 * ，左边显示数据

**二、填空题**

1. 在 ios_base 类中定义的用于控制输入/输出格式的枚举常量中，用于控制对齐方式的 3 个常量的名字是_____。

2. 表达式 cout<<hex 还可表示为_____。

3. 在 ios_base 类中定义的用于控制输入/输出格式的枚举常量中，用于控制浮点数表示形式（科学计数法、定点表示法）的 2 个常量的名字是_____。

4. 表达式 cout<<'\n'还可表示为_____。

5. 下面程序的输出结果是

```
****987.65
***1234.56
```

请将程序中遗漏部分补充完整：

```
#include<iostream>
#include<iomanip>
using namespace std;
int main(){
    cout.setf(ios::fixed);
    cout._____;
    cout.fill('*');
```

```
    cout.width(10);
    cout<<987.6543<<endl<<_______<<1234.5678;
    return 0;
}
```

6. 下面程序的输出结果是：

```
5.00000
5
+5
```

请将程序中遗漏部分补充完整：

```
#include<iostream>
#include<iomanip>
using namespace std;
int main(){
      double x=5;
      cout<<_______ <<x;
      cout<<endl<<_______<<x;
      cout<<endl<< _______ <<x;
      return 0;
}
```

7. 下面程序的输出结果是：

```
TTTTTTTTT5.23
5.23TTTTTTTTT
```

请将横线上遗漏部分补充完整：

```
#include<iostream>
#include<iomanip>
using namespace std;
int main(){
      double  i=5.23;
      cout<< _______<<setw(14)<<i;
      cout<<endl<<_________<<setw(_______) <<i;
      return 0;
```