

## C++读取文本文件的几种方法

---

前几天要用到C++读取文本文件，就学习了一下几种不同的读取方法：

文本文件内容如下：

```
0 1 2 3 4 5 6 7 8 9
0 1 2 3 4 5 6 7 8 9
0 1 2 3 4 5 6 7 8 9
0 1 2 3 4 5 6 7 8 9
```

第一种：直接读取，以空格换行

```
int main ()
{
    ifstream infile;
    infile.open("qqz1.txt", ios::in);
    if (!infile.is_open())
    {
        cout << "读取文件失败" << endl;
        return;
    }
    //第一种读取方法，
    char buf[1024] = { 0 };
    while (infile>>buf)
    {
        cout << buf << endl;//输出读取的文本文件数据
    }
}
```

第二种：数组方法，逐行读取，可读取空格

```
int main ()
{
    ifstream infile;
    infile.open("qqz1.txt", ios::in);
    if (!infile.is_open())
    {
        cout << "读取文件失败" << endl;
        return;
    }
}
```

```
//第二种读取方法
char buf[1024];
while (infile.getline(buf,sizeof(buf)))
{
    cout << buf << endl;
}
}
```

第三种：字符串读取，逐行读取，可读取空格

```
int main ()
{
    ifstream infile;
    infile.open("qqz1.txt", ios::in);
    if (!infile.is_open())
    {
        cout << "读取文件失败" << endl;
        return;
    }
    //第三种读取方法
    string buf;
    while (getline(infile,buf))
    {
        cout << buf << endl;
    }
}
```

第四种：逐字符读取，可读取空格，但是效率较低

```
int main ()
{
    ifstream infile;
    infile.open("qqz1.txt", ios::in);
    if (!infile.is_open())
    {
        cout << "读取文件失败" << endl;
        return;
    }
    //第四种读取方法
    char c;
    while ((c=infile.get())!=EOF)
    {
        cout << c;
    }
}
```

```
}  
}
```

### 第五种：读取至Vector容器中

```
int main ()  
{  
    ifstream infile;  
    infile.open("qqz1.txt", ios::in);  
    if (!infile.is_open())  
    {  
        cout << "读取文件失败" << endl;  
        return;  
    }  
    //第五种读取方法  
    string s;  
    vector<string>v1;  
    while (getline(infile,s))  
    {  
        infile >> s;  
        v1.push_back(s);  
    }  
    for (int i = 0; i < v1.size(); i++)  
    {  
        cout << v1.at(i);  
        cout << endl;  
    }  
    infile.close();  
}
```

这是暂时总结的几种方法，就当记录一下。