1,从键盘中输入一串仅由英文字母组成的字符串，统计26个字母出现的频率，不区分大小写，按频率从大到小输出。

2，从文本文件中读取数据，存放到二进制文件中，从二进制文件中再读取，按要求存放到文本文件下

3，将向量中重复元素去除，并按从大到小输出

4，对12个月份的英文字母进行字典排序

5，向量成员为结构体，根据学号进行排序，删除其中名字为TOM的人，保存到文件中

6,计算整数占多少列的函数

7，二维向量

8，多头文件

1，

#include<iostream>

#include<string>

#include<cmath>

#include<iomanip>

#include<sstream>

#include<vector>

#include<string.h>

using namespace std;

int main()

{

string s;

char a1,hu;

vector<int> a(30,0); //先定义一个向量里面有30个元素,赋初值为0

int i=0,k,t=0;

vector<char> b; //定义一个字符向量

for(i=97;i<=122;i++)

{

b.push\_back(i); //将小写字母a~z的ASIIC码全都赋给向量b

}

cout<<"请输入一个仅有英字母的字符，没有空格,最多200个字符组成"<<endl;

cin>>s;

k=s.size();

for(istringstream sin(s);sin>>a1;)

{

if(a1<97)

{

a1=a1+32; //如果是大写字母转换成小写的

}

for(i=0;i<26;)

{

if(a1==b[i]) //将a1中的字母与向量中的26个小写字母一个一个的相比较，如果一样，就将对应的向量中D的加一并结束本次循环

{

a[i]++;

break;

}

else

i++;

}

i=0; //需要归零的

}

for(int j=0;j<25;j++)

{ //函数是对各个字母出现的次数进行排序

for(int i=0;i<25-j;i++)

{

if(a[i]<a[i+1])

{

t=a[i];

a[i]=a[i+1];

a[i+1]=t;

hu=b[i];

b[i]=b[i+1];

b[i+1]=hu;

}

}

}

for(i=0;i<26;i++)

{

if(a[i]==0)

{

continue;

}

else

{

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

}

}

return 0;

}

,

2，

#include<iostream>

#include<fstream>

#include<iomanip>

#include<string>

#include<sstream>

using namespace std;

int main()

{

ifstream in("Data.txt");

ofstream out("Res.dat",ios::binary);

ifstream in1("Res.dat",ios::binary);

ofstream out1("Res.txt");

int temp;

int tem;

while(in>>temp) //从Data.txt文件中读取数据，将读到的数据一个一个的输给整形变量temp

{

out.write((char\*)&temp,sizeof(int)); //将整型变量中的数据写到二进制文件中去¤

}

in.close();

out.close(); // 每次对文件操作完以后要将文件关闭

for(int i=1;in1.read((char\*)&tem,sizeof(int));i++) //从二进制文件中读取数据

{

if(i%2==0)

{

out1<<setw(10)<<setfill('c')<<right<<tem<<endl;

}

else

out1<<setw(10)<<setfill('c')<<right<<tem;

}

in1.close();

out1.close(); //每次对文件操作完以后要将文件关闭

return 0;

}

3，

#include<iostream>

#include<string>

#include<iomanip>

#include<vector>

#include<time.h>

#include<functional> //调用一些函数

#include<algorithm> //里面包含一些函数

using namespace std;

int main()

{

int a[]={35,46,57,13,24,35,68,13,79,88,46};

vector<int> b(a,a+sizeof(a)/sizeof(int));

sort(b.begin(),b.end(),less<int>()); //将向量排序

int i;

for(i=0;i<b.size()-1;i++) //如果有重复的数，就将重复的赋以这个比较小的值

{

if(b[i]==b[i+1])

{

b[i]=-1;

}

}

sort(b.begin(),b.end(),less<int>()); //再次进行排序

for(i=0;i<b.size();i++) //输出所有的正整数

{

if(b[i]<0)

{

continue;

}

else

{

cout<<b[i]<<endl;

}

}

return 0;

}

4，

#include<algorithm>

#include<iostream>

#include<fstream>

#include<string>

#include<cmath>

#include<iomanip>

#include<vector>

#include<deque>

#include<functional>

using namespace std;

int main()

{

vector<string> str;

str.push\_back("January");

str.push\_back("February");

str.push\_back("March");

str.push\_back("April");

str.push\_back("May");

str.push\_back("June");

str.push\_back("July");

str.push\_back("Aguest");

str.push\_back("September");

str.push\_back("October");

str.push\_back("November");

str.push\_back("December"); //向向量的最后添加元素及12个月份的英文字母

sort(str.begin(),str.end(),less<string>()); //利用排序函数，对12个月份的英文字符串进行按照字典序的排序

for(unsigned int i=0;i<str.size();i++)

{

cout<<str[i]<<endl; //将string向量中的元素输出

}

return 0;

}

5

#include<iostream>

#include<iomanip>

#include<string>

#include<fstream>

#include<vector>

#include<functional>

#include<algorithm>

using namespace std;

struct Student //定义结构体

{

int ID; //学号

string name; //姓名

float score[3]; //三门成绩的分数

};

/\*功能：排序

\*/

void Sort( vector<struct Student>&a) //定义按照学号排序的函数

{

int i=0,j=0;

for( i=0;i<a.size()-1;i++)

{

for( j=0;j<a.size()-1-i;j++)

{

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

{

Struct Student t;

t=a[j];

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

a[j+1]=t;

}

}

}

}

/\*功能：删除向量中符合条件的元素并将处理好的向量写到一个文件中去

\*/

void deletandwrite(vector<struct Student>&a,string fn)

{

int i,j;

ofstream out(fn);

for(i=0,j=0;i<a.size();i++)

{

if(a[i].name=="TOM")

{

continue;

}

else

{

a[j].ID=a[i].ID;

a[j].name=a[i].name;

a[j].score[0]=a[i].score[0];

a[j].score[1]=a[i].score[1];

a[j].score[2]=a[i].score[2];

out<<setw(10)<<right<<a[j].ID<<setw(10)<<right<<a[j].name<<setprecision(3)<<right<<a[j].score[0]<<setprecision(3)<<right<<a[j].score[1]<<setprecision(3)<<right<<a[j].score[2]<<endl;

j++;

}

}

}

/\*功能：将一个向量给显示在屏幕上

\*/

void display( vector<struct Student>&a)

{

for(unsigned int i=0;i<a.size();i++)

{

cout<<right<<setw(10)<<a[i].ID<<setw(10)<<right<<a[i].name;

cout<<setprecision(3)<<right<<showpoint<<a[i].score[0]<<setprecision(3)<<right<<a[i].score[1]<<setprecision(3)<<right<<a[i].score[2]<<endl;

}

}

int main()

{

vector<struct Student> a;

struct Student b;

for(int i=0;i<5;i++)

{

cout<<"请输入第"<<i+1<<"个学生的学号，姓名，三门成绩的分数y："<<endl;

cin>>b.ID;

cin>>b.name;

cin>>b.score[0];

cin>>b.score[1];

cin>>b.score[2];

a.push\_back(b); //向向量中插入数据

}

Sort(a);

display(a);

deletandwrite(a,"d:\\res.dat");

return 0;

}

6

void width(int n)

{

int i=0;

while( n > 0 )

{

i++; //余数是最后一位数

n /= 10; //将最后一位去掉

}

cout<<"这个整数所占的宽度是："<<i<<endl;

}

7，

#include<iostream>

#include<fstream>

#include<sstream>

#include<string>

#include<vector> //向量的库

#include<iomanip>

using namespace std;

int main()

{

ifstream in( "a.txt" );

int element;

vector< vector<int> > m; //构成元素是整数向量，一开始中也是什么都没有的

vector<int> line; //定义以个向量line ,是由整型元素构成的，一开始也是里面什么也没有

for ( string s; getline( in, s ); ) /一行一行的将文件中的内容读出来赋给

{

line.clear(); //将line向量中元素清空

for ( istringstream sin( s ); sin >> element; line.push\_back( element ) ); //将string流中输入到整数中，将每次读到的整数插到向量line的最后

m.push\_back( line ); //将向量line 插入到向量的最后

}

in.close();

for ( unsigned int i = 0; i < m.size(); i++ ) // 控制行

{

for ( unsigned int j = 0; j < m[i].size(); j++ ) //控制列

{

cout << setw( 5 ) << left << m[i][j];

}

cout << endl;

}

return 0;

}

8，

01.cpp

#include<iostream>

#include"name1.h"

#include"name2.h"

using namespace std;

using namespace n1;

using namespace n2;

//using namespace n3;

int main()

{

f1();

f2();

n3::f2();

return 0;

}

Name1.cpp

#include <iostream>

#include "name1.h"

using namespace std;

namespace n1

{

void f1()

{

cout<<"this is n1::f1();"<<endl;

return;

}

}

Name1.h

namespace n1

{

void f1();

}

Name2.cpp

#include <iostream>

#include "name2.h"

using namespace std;

namespace n2

{

void f2()

{

cout<<"this is n2::f2();"<<endl;

return;

}

}

namespace n3

{

void f2()

{

cout<<"this is n3::f2();"<<endl;

return;

}

}

Name2.h

namespace n2

{

void f2();

}

namespace n3

{

void f2();

}