**9921103163**

**Nitin Chaudhary**

**F8**

**Week 10**

**Ans 1-**

#include<iostream>

#include<map>

using namespace **std**;

int **main**()

{

**map**<int,**string**> Students;

Students**[**200**]=**"Alice";

Students.**insert**({201,"John"});

int size = Students.**size**();

cout**<<**"Size of map: "**<<**size**<<endl**;

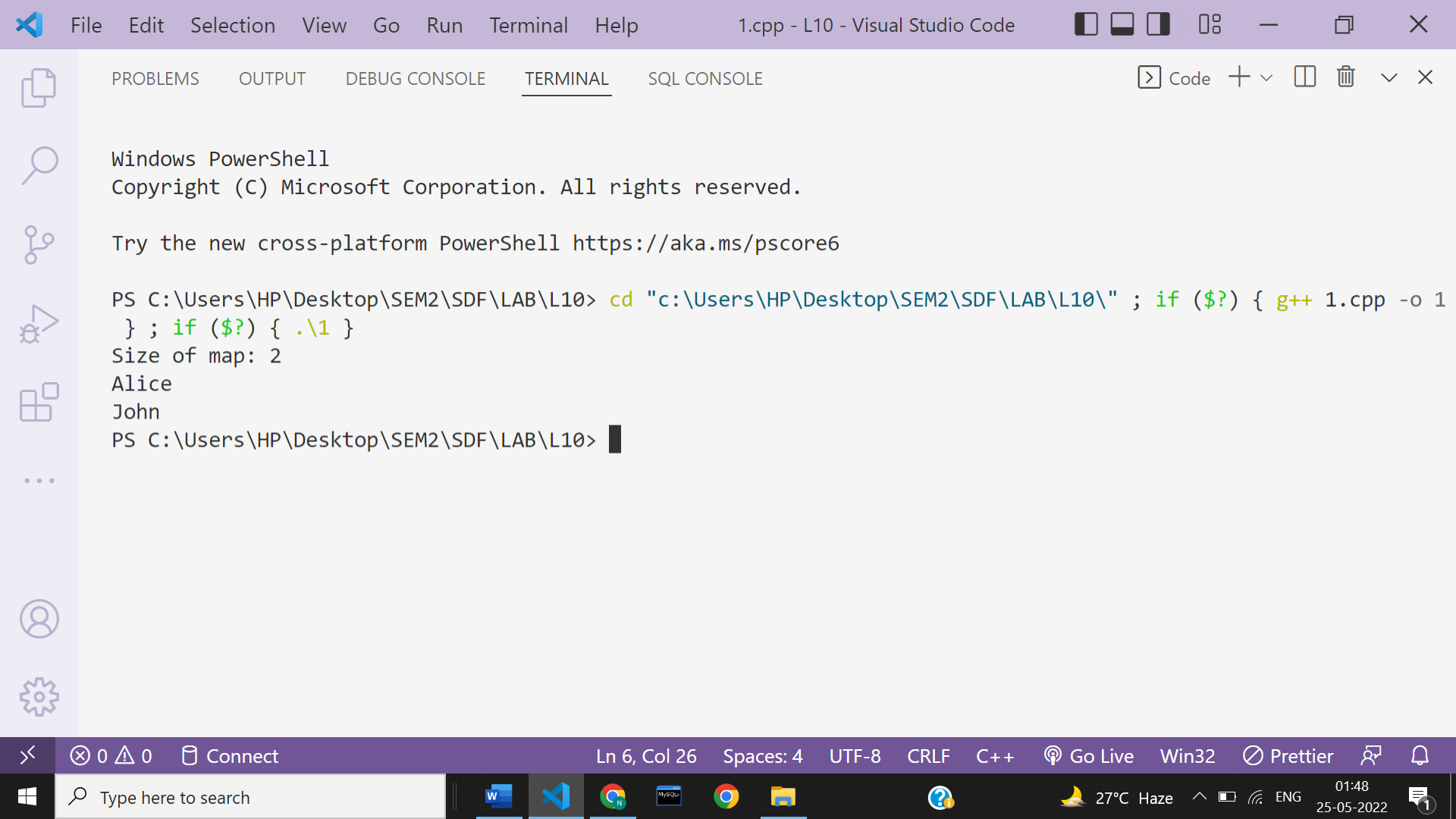
**map**<int,**string**> ::**iterator** itr;

for(itr **=** Students.**begin**(); itr**!=** Students.**end**();itr**++**){

cout**<<**itr**->**second**<<endl**;

}

}



**Ans 2-**

#include<iostream>

#include<map>

using namespace **std**;

int **main**()

{

**map**<int,int> m;

m.**insert**({7,6});

m.**insert**({4,800});

m.**insert**({9,3});

**map**<int,int>::**iterator** itr;

int k,v;

cout**<<**"Enter the key: ";

cin**>>**k;

if(m.**count**(k)){

cout**<<**k**<<**" is already present in the map"**<<endl**;

for(itr **=** m.**begin**(); itr**!=** m.**end**();itr**++**){

if(itr**->**first == k){

cout**<<**itr**->**first**<<**"->"**<<**itr**->**second**<<endl**;

}

}

}

else{

cout**<<**"Enter the value: ";

cin**>>**v;

m.**insert**({k,v});

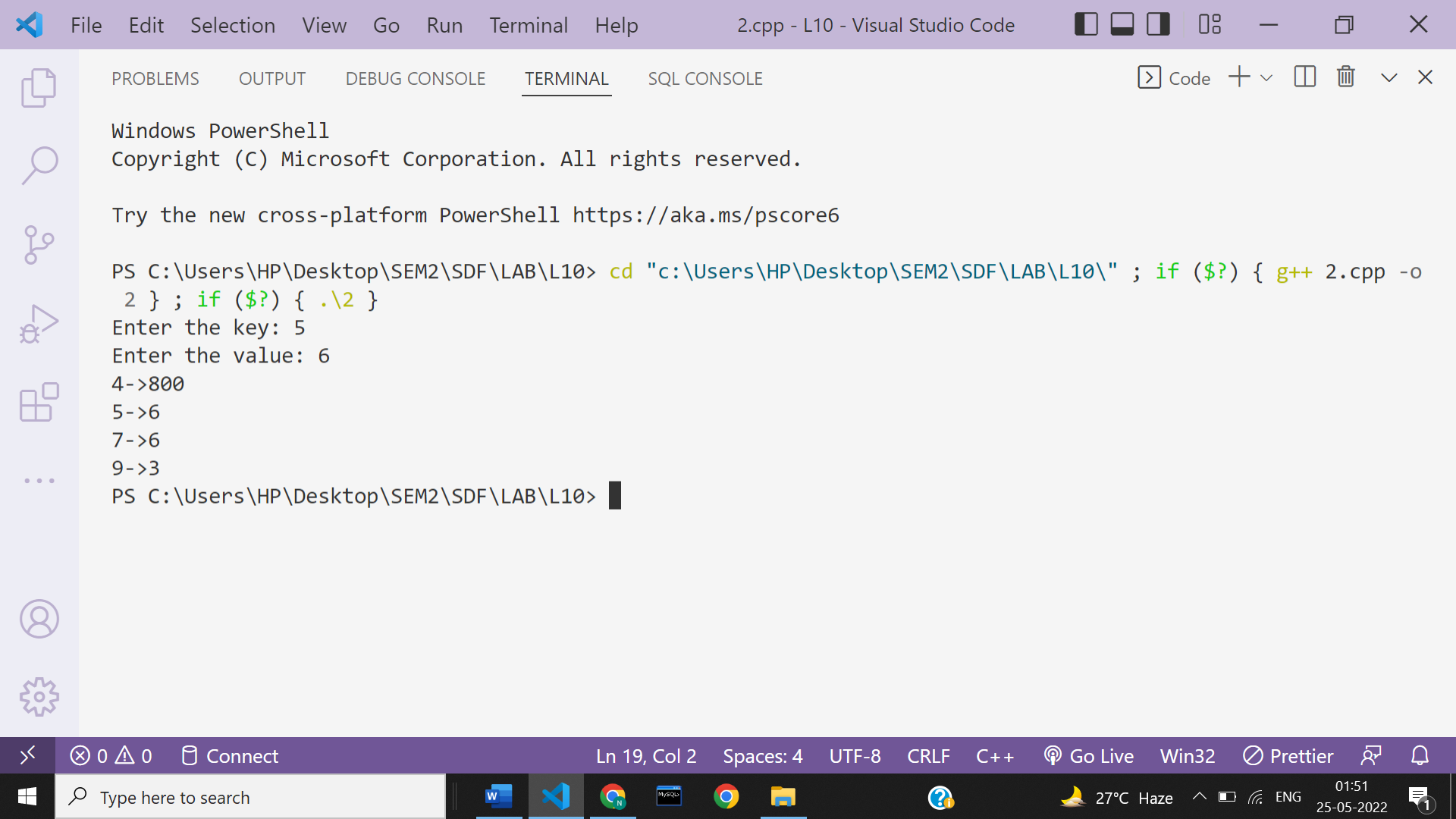
for(itr **=** m.**begin**(); itr**!=** m.**end**();itr**++**){

cout**<<**itr**->**first**<<**"->"**<<**itr**->**second**<<endl**;

}

}

}



**Ans 3-**

#include<iostream>

#include<map>

using namespace **std**;

int **main**()

{

**map**<int,**string**> Students;

Students**[**200**]=**"Alice";

Students.**insert**({201,"John"});

**map**<int,**string**> ::**iterator** itr;

int x;

cout**<<**"Enter the key to be searched: ";

cin**>>**x;

itr **=** Students.**find**(x);

if(itr**->**first == x){

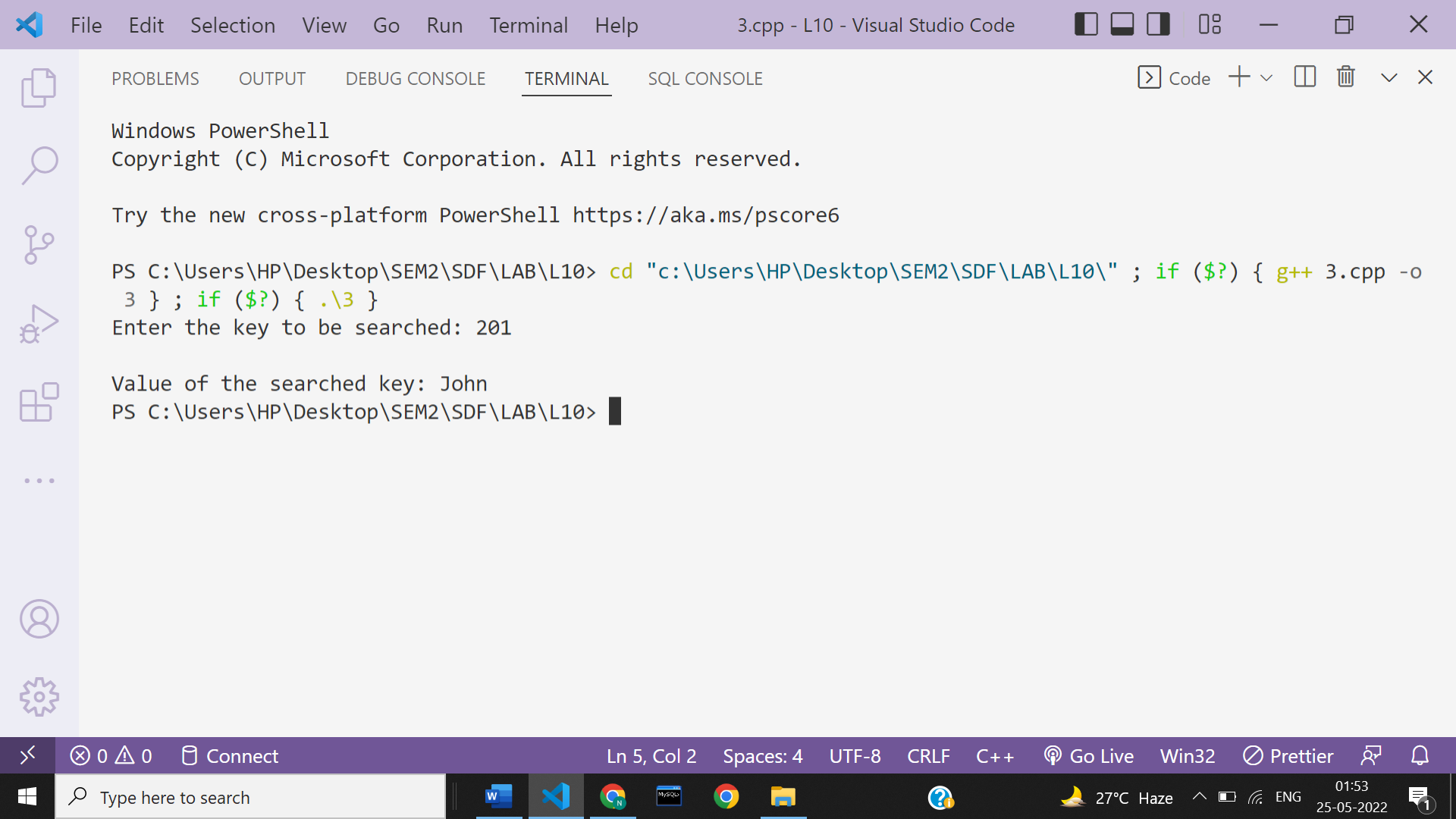
cout**<<**"\nValue of the searched key: "**<<**itr**->**second**<<endl**;

}else{

cout**<<**"Key not found";

}

}



**Ans 4-**

#include<iostream>

#include<map>

using namespace **std**;

int **main**()

{

**map**<**string**,int> my\_map;

my\_map.**insert**({"Cow",1});

my\_map.**insert**({"Cat",2});

my\_map.**insert**({"Lion",3});

**map**<**string**,int>::**iterator** itr;

**map**<**string**,int>::**iterator** del;

del **=** my\_map.**find**("Cat");

if(del **!=**my\_map.**end**()){

my\_map.**erase**(del);

}

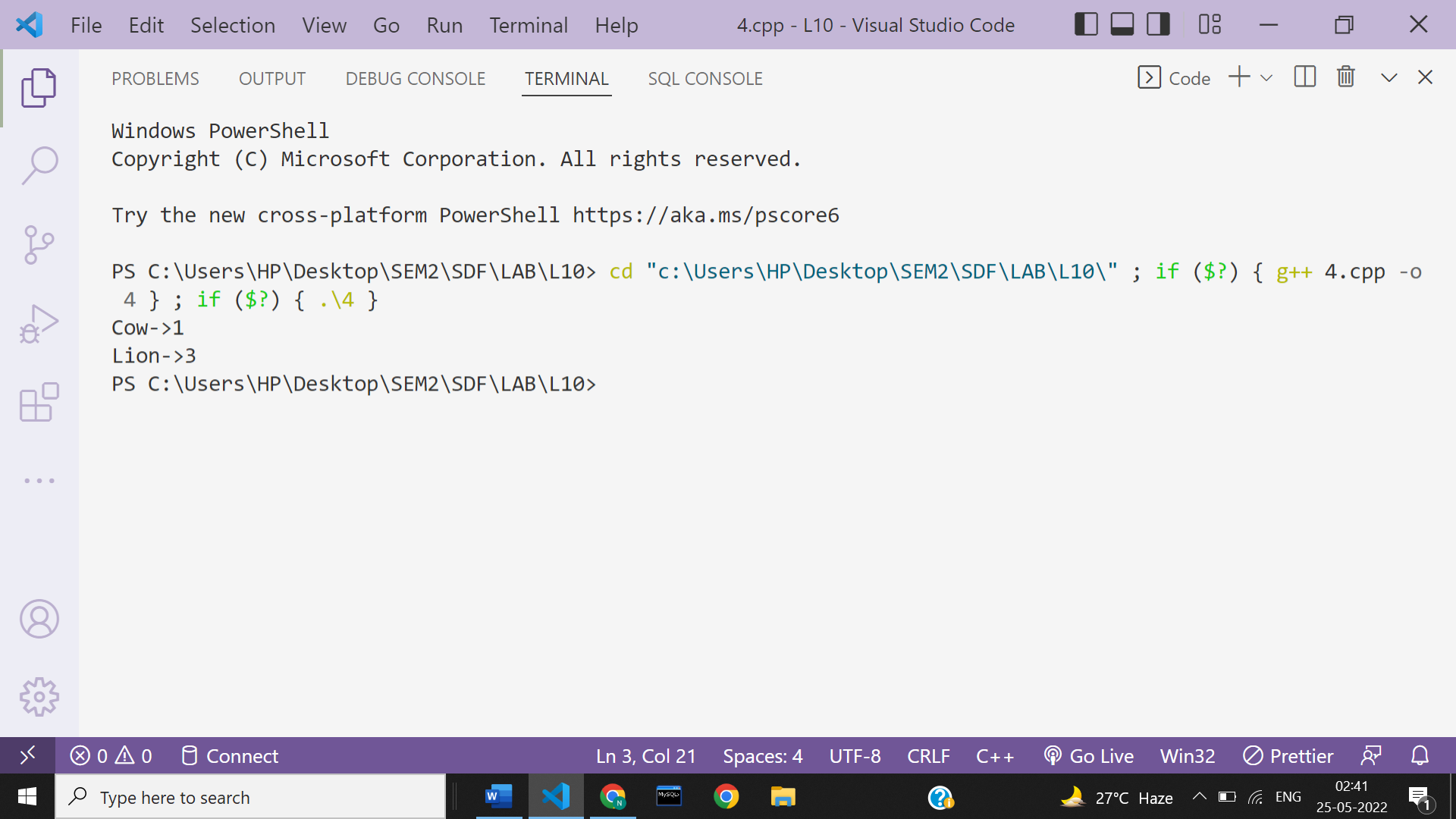
*//Elements in map after deletion*

for(auto itr=my\_map.**begin**();itr **!=** my\_map.**end**();itr**++**){

cout**<<**itr**->**first**<<**"->"**<<**itr**->**second**<<endl**;

}

}



**Ans 5-**

#include<iostream>

#include<fstream>

#include<string>

using namespace **std**;

**main**()

{

**ofstream** myfile;

myfile.**open**("Data.txt");

**string** str;

cout**<<**"Enter the data: ";

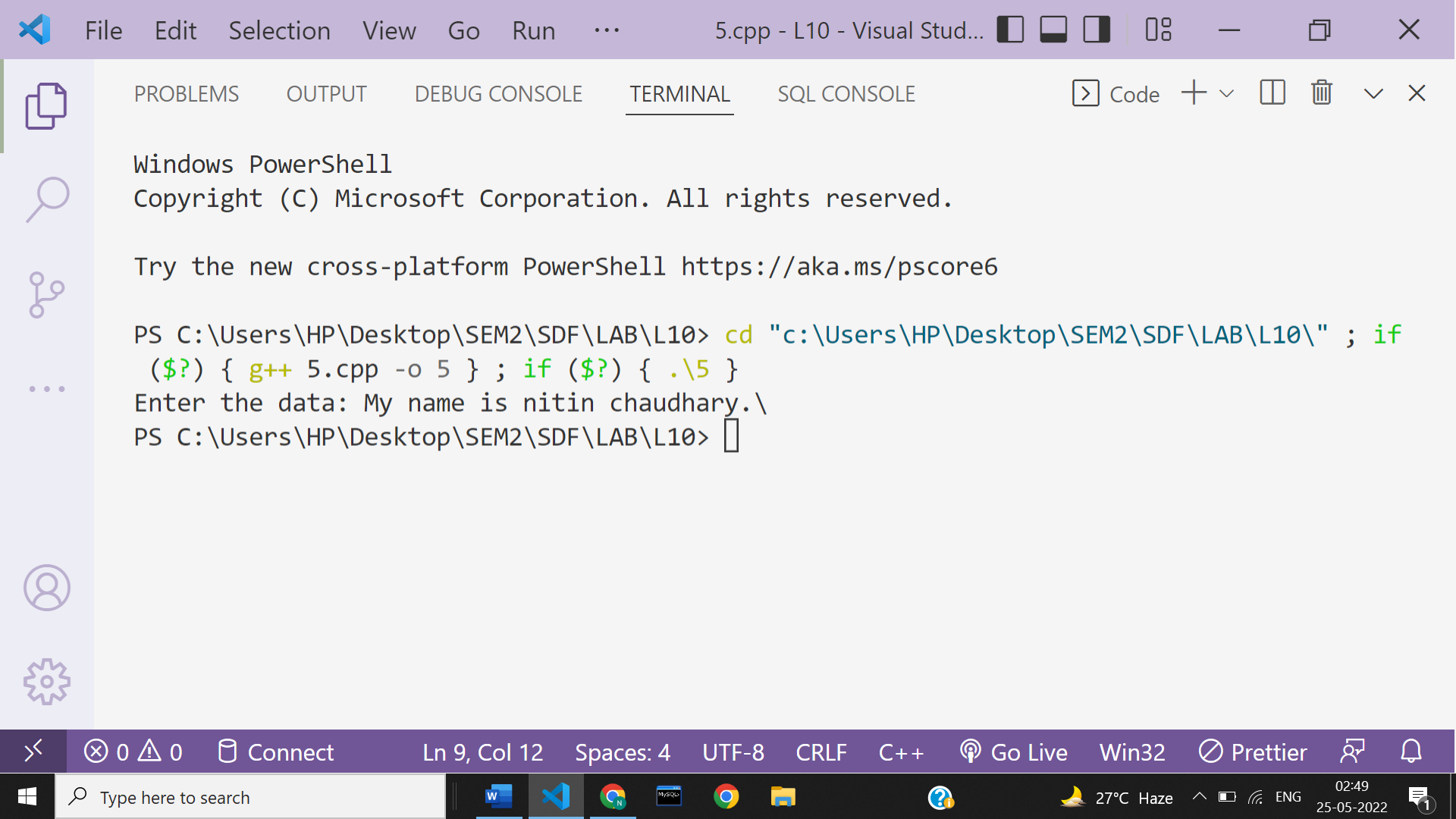
**getline**(cin,str);

*//myfile.write(str.c\_str(),str.length());*

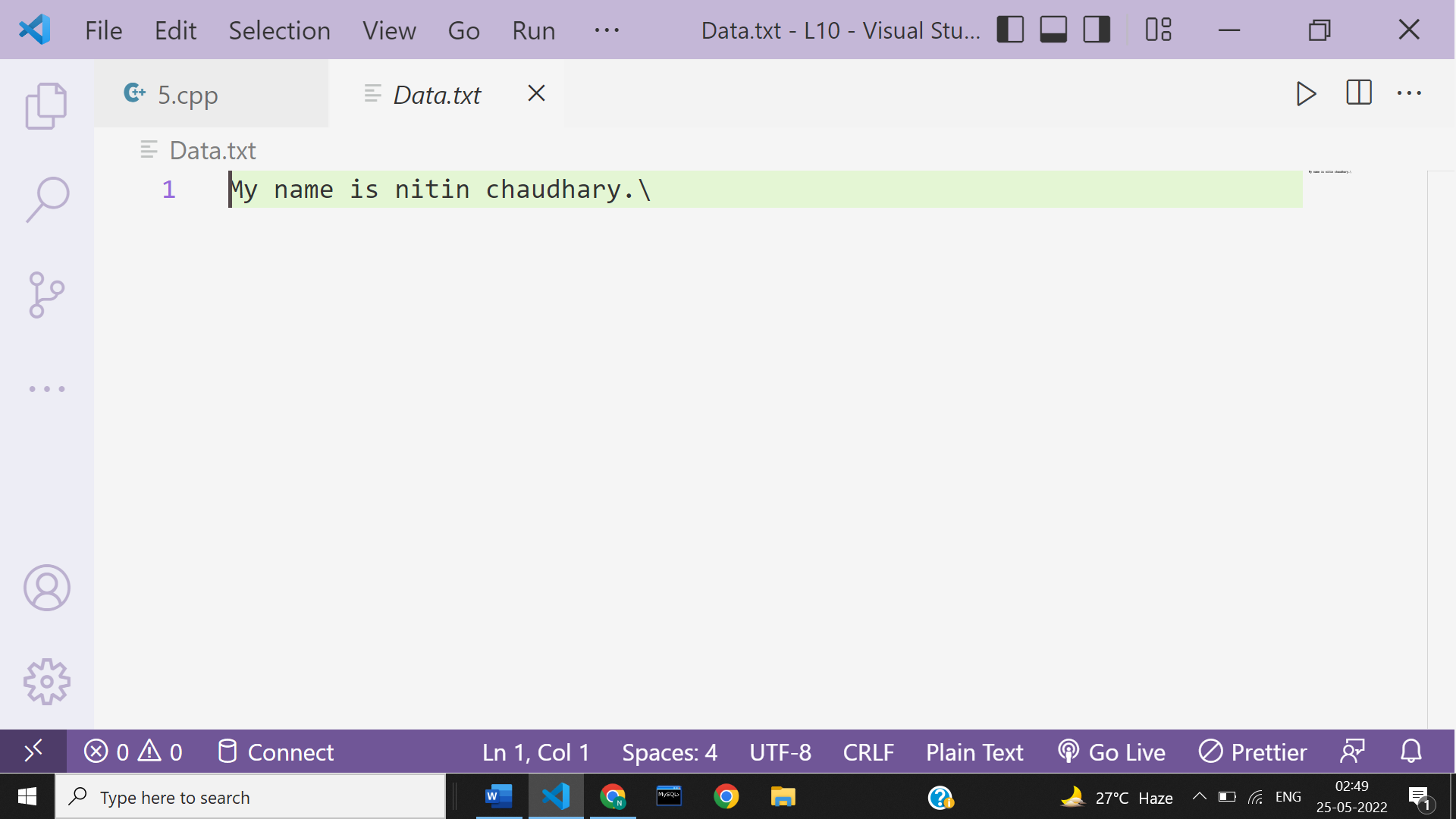
myfile**<<**str;

myfile.**close**();

}



**Data in file-**



**Ans 6-**

#include<iostream>

#include<fstream>

#include<string>

using namespace **std**;

**main**()

{

**fstream** myfile;

myfile.**open**("Data.txt",**ios**::in);

**string** str;

while(myfile.**eof**()==0){

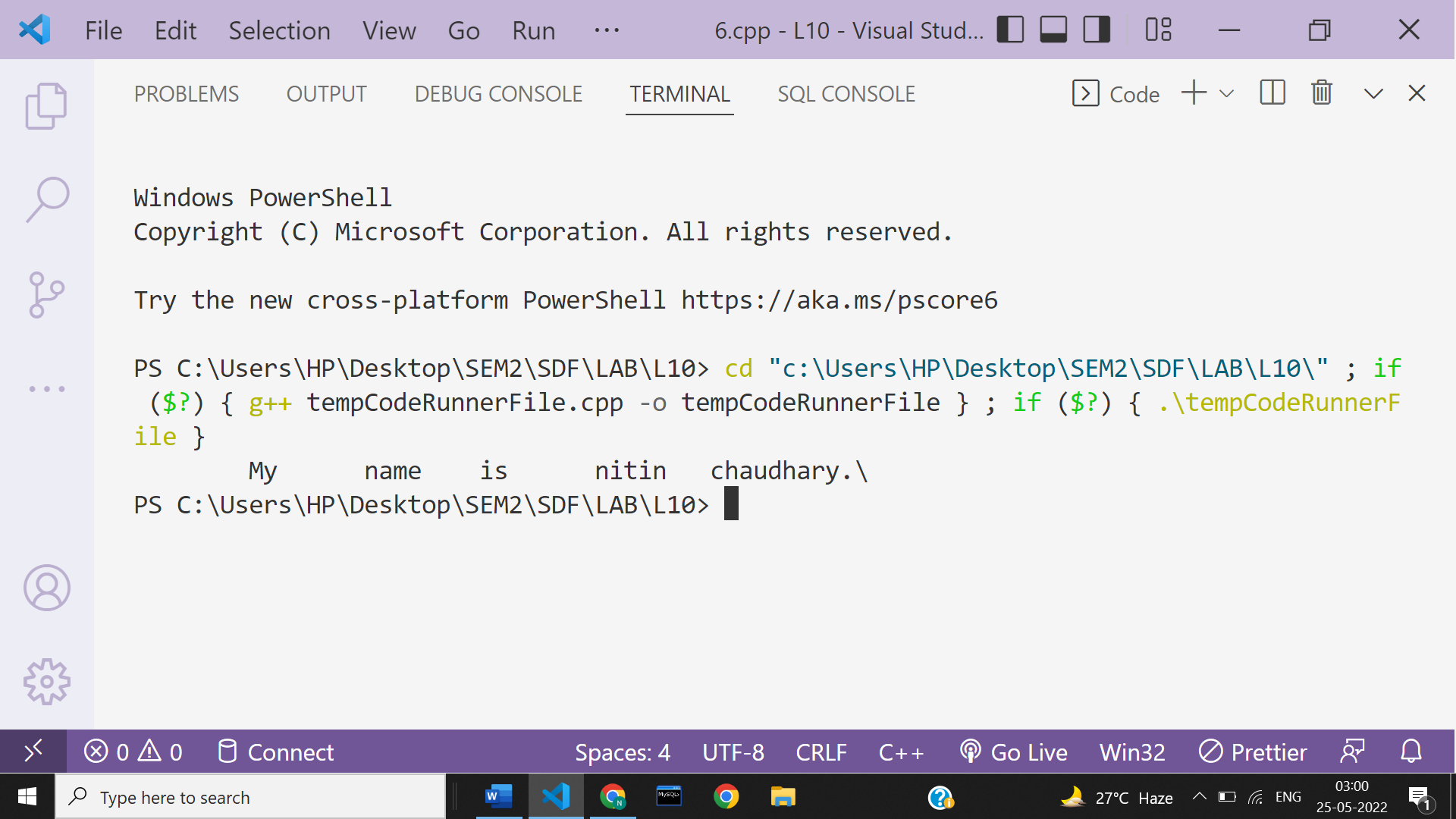
myfile**>>**str;

cout**<<**"\t"**<<**str;

}

myfile.**close**();

}



**Ans 7-**

#include <iostream>

#include <string>

#include <fstream>

using namespace **std**;

int **main**()

{

**ifstream** sf;

**ofstream** df;

**string** str;

**string** sourcefile, destinationfile;

cout **<<** "Enter Source File with Extension: ";

**getline**(cin, sourcefile);

sf.**open**(sourcefile);

if (**!**sf)

{

cout **<<** "Error in Opening Source File...!!!";

**exit**(1);

}

cout **<<** "Enter Destination File with Extension: ";

**getline**(cin, destinationfile);

df.**open**(destinationfile);

if (**!**df)

{

cout **<<** "Error in Opening Destination File...!!!";

sf.**close**();

**exit**(2);

}

if (sf && df)

{

while (**getline**(sf, str))

{

df **<<** str **<<** "\n";

}

cout **<<** "\n\n Source File Date Successfully Copied to Destination File...!!!";

}

else

{

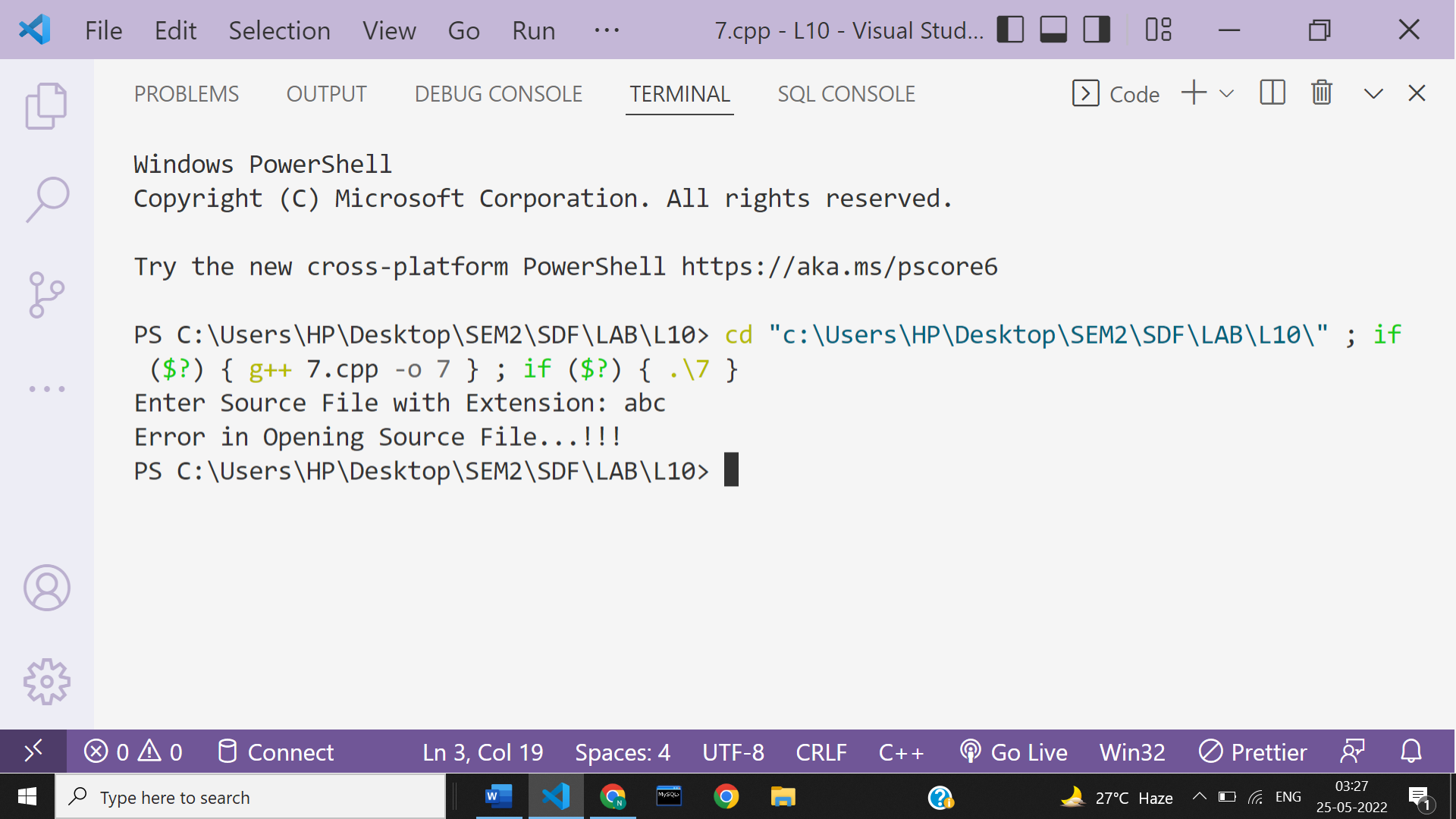
cout **<<** "File Cannot Open...!!!";

}

sf.**close**();

df.**close**();

}



**Ans 8-**

#include <iostream>

#include <string>

#include <fstream>

using namespace **std**;

int **main**()

{

**fstream** sf,sf2,df;

**string** str1,str2;

**string** source1,source2, destinationfile;

cout **<<** "Enter first Source File with Extension: ";

**getline**(cin, source1);

sf.**open**(source1,**ios**::in);

if (**!**sf)

{

cout **<<** "Error in Opening Source File...!!!";

**exit**(1);

}

cout **<<** "\nEnter second Source File with Extension: ";

**getline**(cin, source2);

sf2.**open**(source2,**ios**::in);

if (**!**sf2)

{

cout **<<** "Error in Opening Source File...!!!";

**exit**(1);

}

cout **<<** "\nEnter Destination File with Extension: ";

**getline**(cin, destinationfile);

df.**open**(destinationfile,**ios**::out);

if (**!**df)

{

cout **<<** "Error in Opening Destination File...!!!";

sf.**close**();

**exit**(2);

}

if (sf && sf2 && df)

{

while (**getline**(sf, str1))

{

df **<<** str1 **<<** **endl**;

}

while (**getline**(sf2,str2))

{

df **<<** str2 **<<endl**;

}

cout **<<** "\n\n Source File Date Successfully Copied to Destination File...!!!";

}

else

{

cout **<<** "File Cannot Open...!!!";

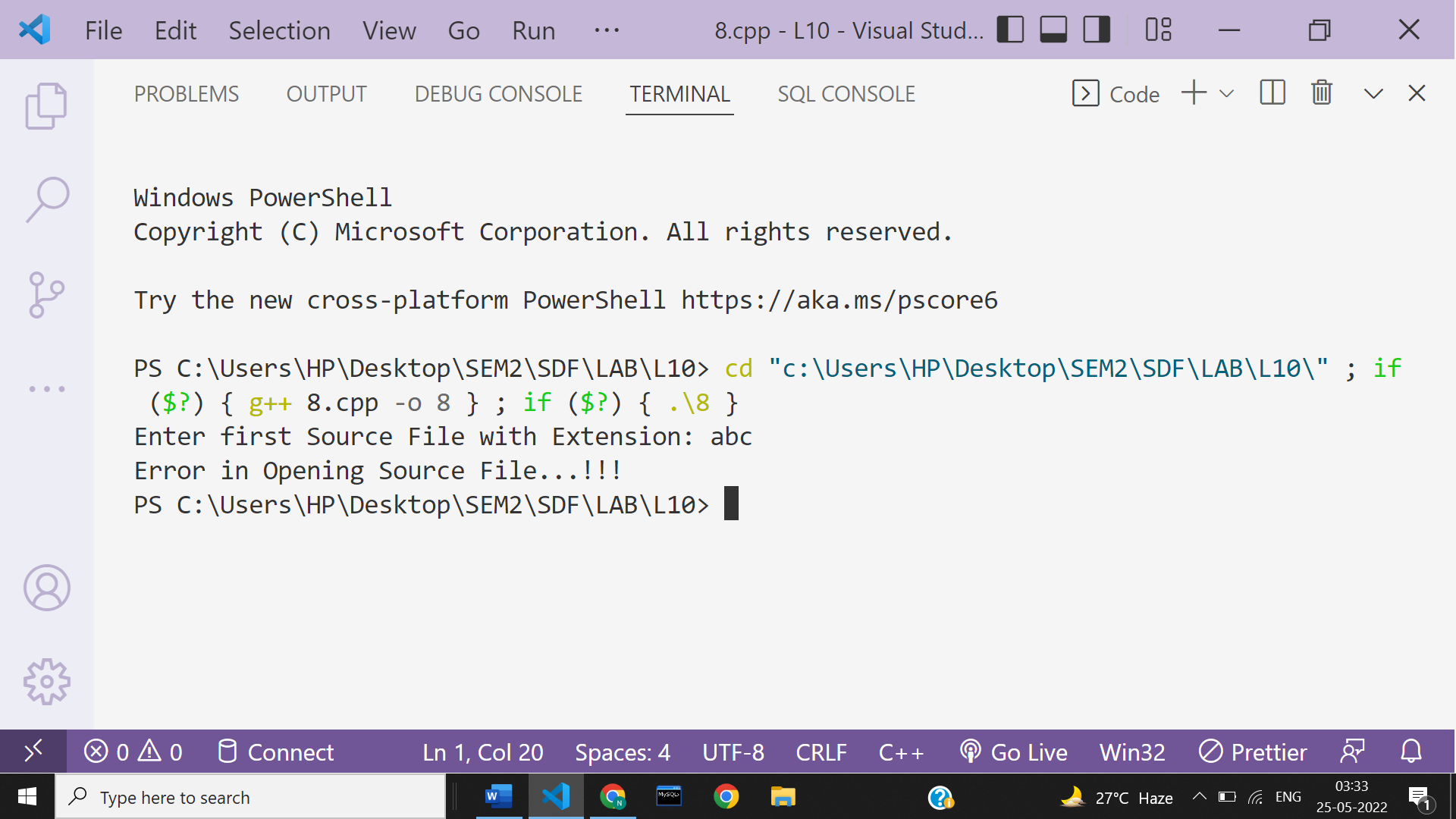
}

sf.**close**();

sf2.**close**();

df.**close**();

}



**Ans 9-**

#include<iostream>

#include<stdio.h>

using namespace **std**;

int **main**()

{

int status;

char fileName[20];

cout**<<**"Enter the Name of File: ";

cin**>>**fileName;

status = **remove**(fileName);

if(status==0)

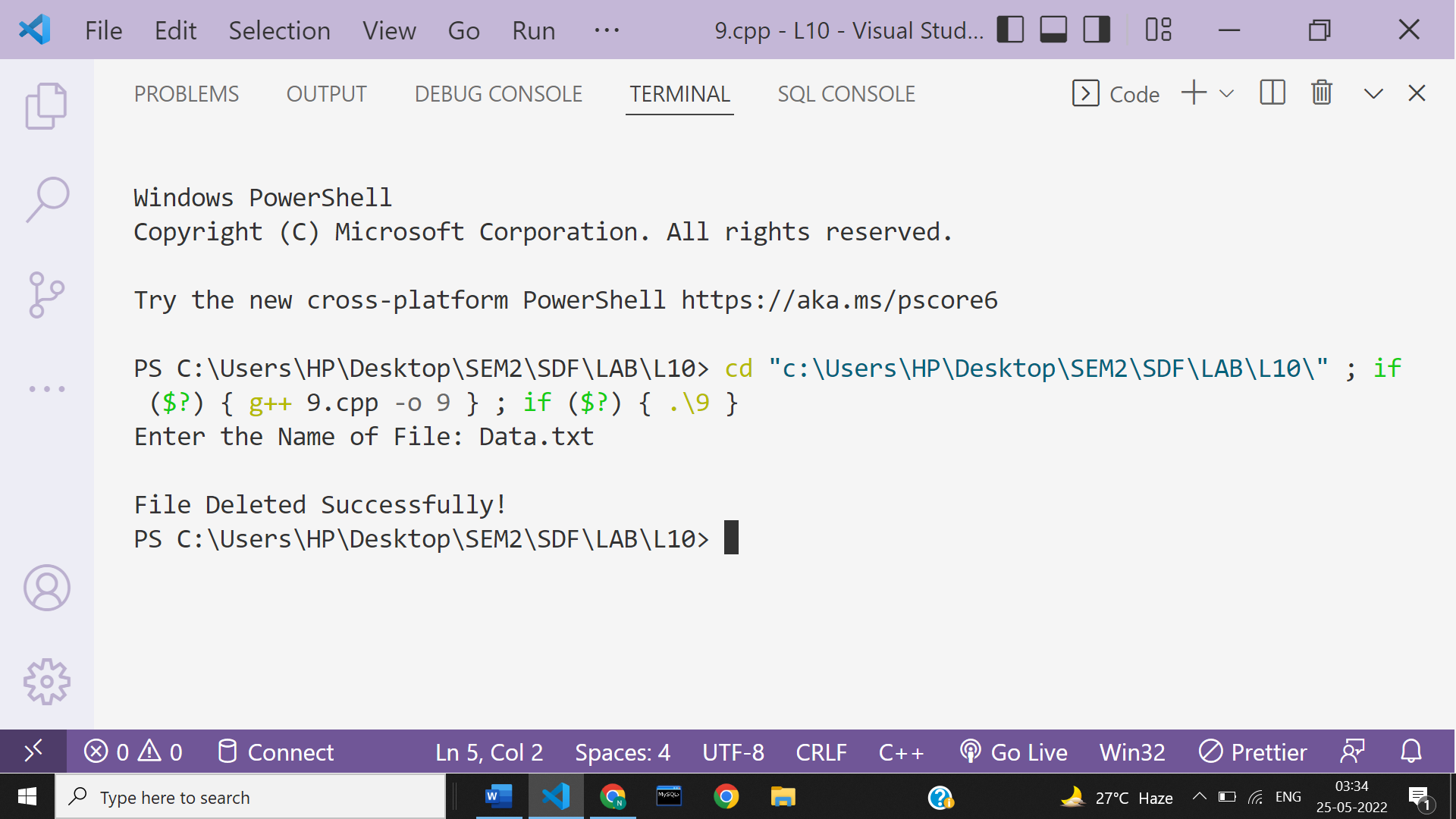
cout**<<**"\nFile Deleted Successfully!";

else

cout**<<**"\nError Occurred!"**<<endl**;

return 0;

}



**Ans 10-**

#include<iostream>

#include<fstream>

#include<string>

using namespace **std**;

**main**()

{

**fstream** myfile , infile;

**string** str,str1;

myfile.**open**("afile.dat",**ios**::out);

cout**<<**"Enter the data: ";

**getline**(cin,str);

try{

myfile**<<** str;

myfile.**close**();

}catch(...){

cout**<<**"Error !"**<<endl**;

}

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

cout**<<**"\nData after reading:\n";

while(infile.**eof**()==0){

**getline**(infile,str1);

cout**<<**"\n"**<<**str1**<<endl**;

}

infile.**close**();

}

