**Code for Assignment 10 A:**

#include<iostream>

#include<fstream>

using namespace std;

int main()

{

    char data[100];

    ofstream outfile;

    outfile.open("Sample1.dat");

    cout<<"\nWriting to the file\n";

    cout<<"\nEnter your name\n";

    cin.getline(data,100);

    outfile<<data<<endl;

    cout<<"\nEnter your age\n";

    cin>>data;

    cin.ignore();

    outfile<<data<<endl;

    outfile.close();

    ifstream infile;

    infile.open("Sample1.dat");

    cout<<"\nReading from the file"<<endl;

    infile>>data;

    cout<<data<<endl;

    infile>>data;

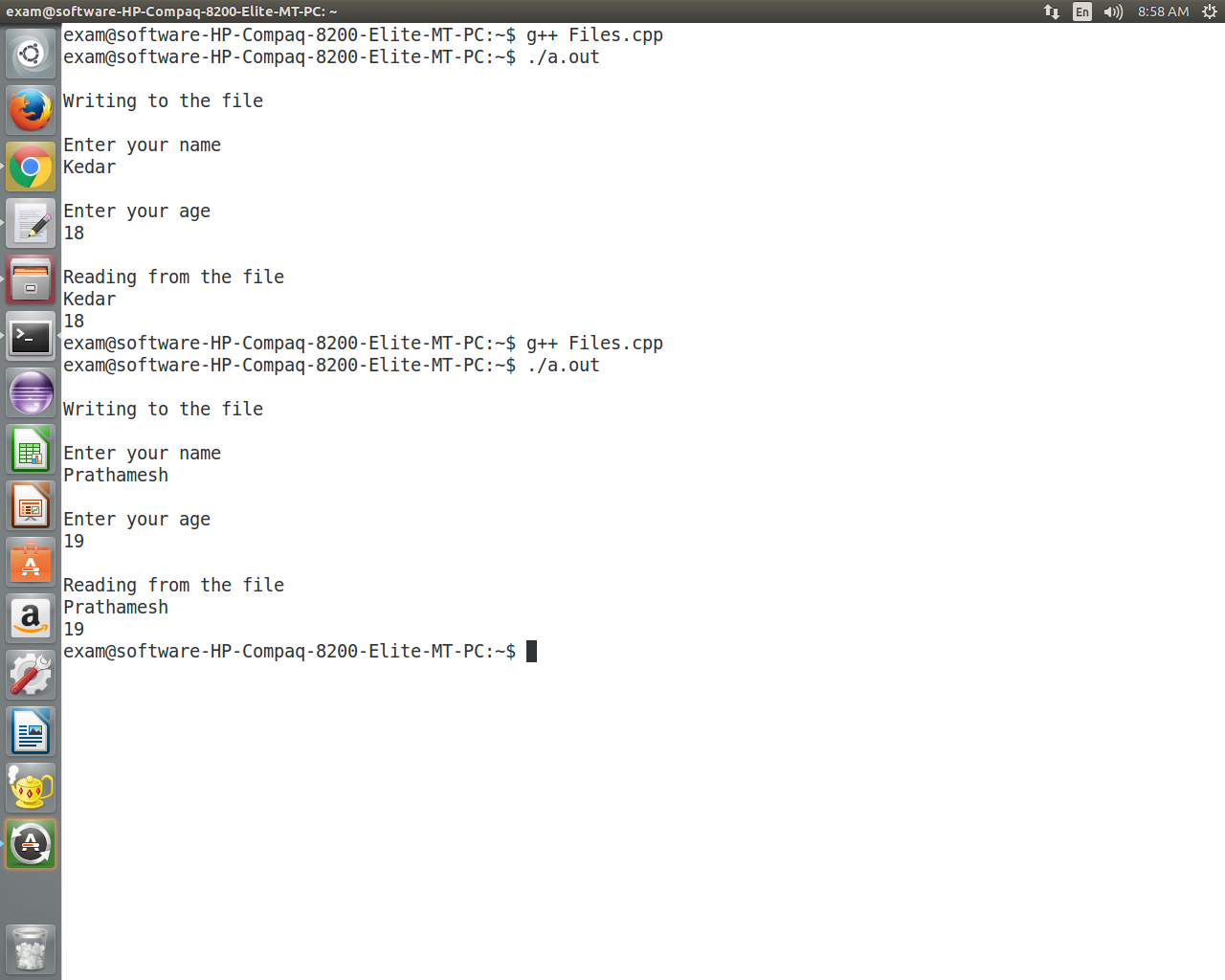
    cout<<data<<endl;

    infile.close();

    return 0;

}

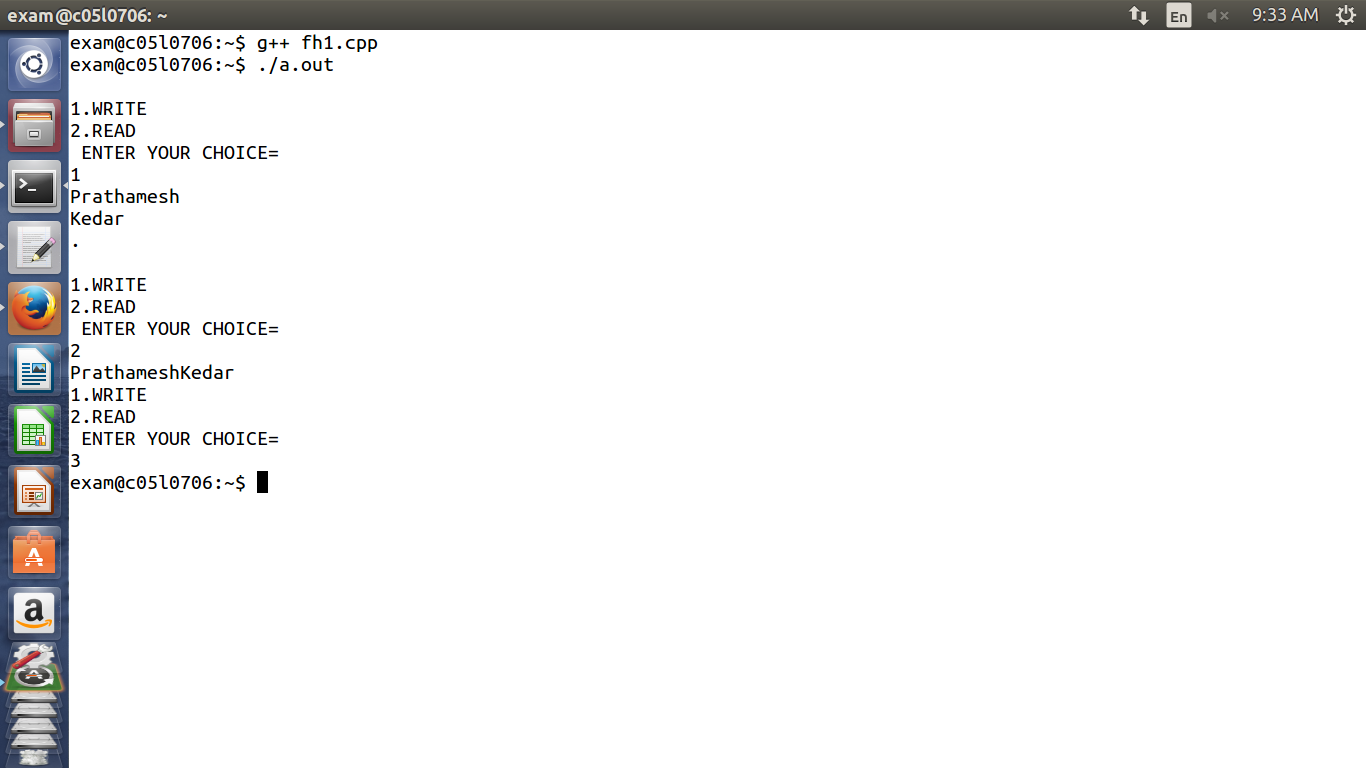
**Output:**

****

**Code for Assignment 10 B:**

#include <iostream>  
#include <fstream>  
  using namespace std;  
    class test  
    {  
        public:  
                void write\_data();  
                void read\_data();  
    };  
    void test::write\_data()  
    {  
        fstream fp;  
        char ch;  
        fp.open("it.txt",ios::out);  
        cin>>ch;  
         while(ch!='.')  
         {  
           fp.put(ch);  
           cin>>ch;  
         }  
         fp.close();  
    }  
    void test::read\_data()  
    {  
        fstream fp;  
        char ch;  
        fp.open("it.txt",ios::in);  
        ch=fp.get();  
        while(!fp.eof())  
        { cout<<ch;  
            ch=fp.get();  
        }  
          fp.close();  
    }  
  int main()  
  { test ob;  
          int ch;   
            do  
            { cout<<"\n1.WRITE\n2.READ";  
                cout<<"\n ENTER YOUR CHOICE= "<<endl;  
                cin>>ch;  
                 switch(ch)  
                        {  
                           case 1:  
                                ob.write\_data();  
                                break;  
                           case 2:  
                                ob.read\_data();  
                                break;  
                        }  
            }while(ch<3);  
        }

**Output:**

****