Project 6

Part A

Header Files

Package.h

#ifndef PACKAGE\_H

#define PACKAGE\_H

#include <string>

using namespace std;

class Package

{

public:

Package(const string &,const string &,const string &,const string &,int,

const string &,const string &,const string &,const string &,int, double,double);

void setSenderName(const string &);

string getSenderName() const;

void setSenderAddress(const string &);

string getSenderAddress() const;

void setSenderCity(const string &);

string getSenderCity() const;

void setSenderState(const string &);

string getSenderState() const;

void setSenderZIP(int);

int getSenderZIP() const;

void setRecipientName(const string &);

string getRecipientName() const;

void setRecipientAddress(const string &);

string getRecipientAddress() const;

void setRecipientCity(const string &);

string getRecipientCity() const;

void setRecipientState(const string &);

string getRecipientState() const;

void setRecipientZIP(int);

int getRecipientZIP() const;

void setWeight(double);

double getWeight() const;

void setCostPerOunce(double);

double getCostPerOunce() const;

double calculateCost() const;

private:

string senderName;

string senderAddress;

string senderCity;

string senderState;

int senderZIP;

string recipientName;

string recipientAddress;

string recipientCity;

string recipientState;

int recipientZIP;

double weight;

double costPerOunce;

};

#endif

classOvernightPackage.h

#ifndef OVERNIGHT\_H

#define OVERNIGHT\_H

#include "Package.h"

class OvernightPackage: public Package

{

public:

OvernightPackage(const string &,const string &,const string &,const string &,int,

const string &,const string &,const string &,const string &,int,double,double,double);

void setOvernightFeePerOunce(double);

double getOvernightFeePerOunce() const;

double calculateCost() const;

private:

double overnightFeePerOunce;

};

#endif

TwoDayPackage.h

#ifndef TWODAY\_H

#define TWODAY\_H

#include "Package.h"

class TwoDayPackage: public Package

{

public:

TwoDayPackage(const string &,const string &,const string &,const string &,int,

const string &,const string &,const string &,const string &,int,double,double,double);

void setFlatFee(double);

double getFlatFee() const;

double calculateCost() const;

private:

double flatFee;

};

#endif

Main Problem

#include<iostream>

#include<iomanip>

#include"Package.h"

#include"TwoDayPackage.h"

#include"classOvernightPackage.h"

using namespace std;

int main()

{

Package package1("Lou Brown","1 Main St","Boston","MA",11111,"Mary Smith","7 Elm St","New York","NY",22222,8.5,.5);

TwoDayPackage package2("Lisa Klein","5 Broadway","Somerville", "MA",33333,"Bob George","21 Pine Road","Cambridge","MA",44444,10.5,0.65,2.0);

OvernightPackage package3("Ed Lewis","2 Oak St","Boston","MA",55555,"Don Kelly","9 Main St","Denver","CO",66666,12.25,0.7,0.25);

cout<<fixed<<setprecision(2);

cout<<"Package1::\n\nSender:\n"<<package1.getSenderName()<<'\n'<<package1.getSenderCity()<<','<<package1.getSenderState()<< ' '

<<package1.getSenderZIP();

cout<<"Package1::\n\nRecipient:\n"<<package1.getRecipientName()<<'\n'<<package1.getRecipientCity()<<','<<package1.getRecipientState()<< ' '

<<package1.getRecipientZIP();

cout<<"\n\nCost:$"<<package1.calculateCost()<<endl;

cout<<"Package2::\n\nSender:\n"<<package2.getSenderName()<<'\n'<<package2.getSenderCity()<<','<<package2.getSenderState()<< ' '

<<package2.getSenderZIP();

cout<<"Package2::\n\nRecipient:\n"<<package2.getRecipientName()<<'\n'<<package2.getRecipientCity()<<','<<package2.getRecipientState()<< ' '

<<package2.getRecipientZIP();

cout<<"\n\nCost:$"<<package2.calculateCost()<<endl;

cout<<"Package3::\n\nSender:\n"<<package3.getSenderName()<<'\n'<<package3.getSenderCity()<<','<<package3.getSenderState()<< ' '

<<package3.getSenderZIP();

cout<<"Package3::\n\nRecipient:\n"<<package3.getRecipientName()<<'\n'<<package3.getRecipientCity()<<','<<package3.getRecipientState()<< ' '

<<package3.getRecipientZIP();cout<<"\n\nCost:$"<<package3.calculateCost()<<endl;

system ("pause");

}

Program Files

Problem A.cpp

#include<iostream>

#include<iomanip>

#include<vector>

#include"Package.h"

#include"TwoDayPackage.h"

#include"OvernightPackage.h"

using namespace std;

int main()

{

vector<Package\*>packages(3);

packages[0]=new Package("Lou Brown","1 Main St","Boston","MA",11111,"Mary Smith","7 Elm St","New York","NY",22222,8.5,.5);

packages[1]=new TwoDayPackage("Lisa Klein","5 Broadway","Somerville","MA",33333,"Bob George","21 Pine Rd","Cambridge","MA",44444,10.5,.65,2.0);

packages[2]=new OvernightPackage("Ed Lewis","2 Oak St","Boston","MA",55555,"Don Kelly","9 Main St","Denver","CO",66666,12.25,.7,.25);

double totalShippingCost=0.0;

cout<<fixed<<setprecision(2);

for(size\_t i=0;i<packages.size();i++)

{

cout<<"Package"<<i+1<<"\n\nSender:\n"

<<packages[i]->getSenderName()<<'\n'

<<packages[i]->getSenderAddress()<<'\n'

<<packages[i]->getSenderCity()<<","

<<packages[i]->getSenderState()<<' '

<<packages[i]->getSenderZIP();

cout<<"\n\nRecipient:\n"

<<packages[i]->getRecipientName()<<'\n'

<<packages[i]->getRecipientAddress()<<'\n'

<<packages[i]->getRecipientCity()<<","

<<packages[i]->getRecipientState()<<' '

<<packages[i]->getRecipientZIP();

double cost=packages[i]->calculateCost();

cout<<"\n\nCost:$"<<cost<<"\n\n";

totalShippingCost+= cost;

}

cout<<"Total shipping cost:$"<<totalShippingCost<<endl;

system("pause");

}

Package.cpp

#include "Package.h"

Package::Package(const string &sName,const string &sAddress,const string &sCity,const string &sState, int sZIP,

const string &rName,const string &rAddress,const string &rCity,const string &rState, int rZIP,double w,double cost)

: senderName(sName),senderAddress(sAddress),senderCity(sCity),senderState(sState),senderZIP(sZIP),

recipientName(rName),recipientAddress(rAddress),recipientCity(rCity),recipientState(rState),recipientZIP(rZIP)

{

setWeight(w);

setCostPerOunce(cost);

}

void Package::setSenderName(const string &name)

{

senderName=name;

}

string Package::getSenderName() const

{

return senderName;

}

void Package::setSenderAddress(const string &address)

{

senderAddress=address;

}

string Package::getSenderAddress() const

{

return senderAddress;

}

void Package::setSenderCity(const string &city)

{

senderCity=city;

}

string Package::getSenderCity() const

{

return senderCity;

}

void Package::setSenderState(const string &state)

{

senderState=state;

}

string Package::getSenderState() const

{

return senderState;

}

void Package::setSenderZIP(int ZIP)

{

senderZIP= ZIP;

}

int Package::getSenderZIP() const

{

return senderZIP;

}

void Package::setRecipientName(const string &name)

{

recipientName=name;

}

string Package::getRecipientName() const

{

return recipientName;

}

void Package::setRecipientAddress(const string &address)

{

recipientAddress=address;

}

string Package::getRecipientAddress() const

{

return recipientAddress;

}

void Package::setRecipientCity(const string &city)

{

recipientCity=city;

}

string Package::getRecipientCity() const

{

return recipientCity;

}

void Package::setRecipientState(const string &state)

{

recipientState=state;

}

string Package::getRecipientState() const

{

return recipientState;

}

void Package::setRecipientZIP(int ZIP)

{

recipientZIP=ZIP;

}

int Package::getRecipientZIP() const

{

return recipientZIP;

}

void Package::setWeight(double w)

{

weight=(w<0.0)? 0.0 : w;

}

double Package::getWeight() const

{

return weight;

}

void Package::setCostPerOunce(double cost)

{

costPerOunce=(cost<0.0)? 0.0 : cost;

}

double Package::getCostPerOunce() const

{

return costPerOunce;

}

double Package::calculateCost() const

{

return getWeight() \* getCostPerOunce();

}

OvernightPackage.cpp

#include "OvernightPackage.h"

OvernightPackage::OvernightPackage(const string &sName,const string &sAddress,const string &sCity,const string &sState, int sZIP,

const string &rName,const string &rAddress,const string &rCity,const string &rState, int rZIP,double w,double cost,double fee)

:Package(sName,sAddress,sCity,sState,sZIP,rName,rAddress,rCity,rState,rZIP,w,cost)

{

setOvernightFeePerOunce(fee);

}

void OvernightPackage::setOvernightFeePerOunce (double overnightFee)

{

overnightFeePerOunce=(overnightFee<0.0) ? 0.0:overnightFee;

}

double OvernightPackage::getOvernightFeePerOunce() const

{

return overnightFeePerOunce;

}

double OvernightPackage::calculateCost() const

{

return getWeight()\*(getCostPerOunce() + getOvernightFeePerOunce() );

}

TwoDayPackage.cpp

#include "TwoDayPackage.h"

TwoDayPackage::TwoDayPackage(const string &sName,const string &sAddress,const string &sCity,const string &sState, int sZIP,

const string &rName,const string &rAddress,const string &rCity,const string &rState, int rZIP,double w,double cost,double fee)

:Package(sName,sAddress,sCity,sState,sZIP,rName,rAddress,rCity,rState,rZIP,w,cost)

{

setFlatFee(fee);

}

void TwoDayPackage::setFlatFee(double fee)

{

flatFee=(fee<0.0) ? 0.0:fee;

}

double TwoDayPackage::getFlatFee() const

{

return flatFee;

}

double TwoDayPackage::calculateCost() const

{

return Package::calculateCost()+getFlatFee();

}

Part B

Problem B.cpp

#include<iostream>

using namespace std;

template <typename T>

bool isEqualTo(const T & arg1,const T & arg2)

{

return arg1==arg2;

}

class Complex

{

friend ostream & operator<<(ostream &,Complex &);

public:

Complex(int realPart,int iPart):real(realPart),imaginary(iPart)

{

}

bool operator==(const Complex &right) const

{

return real == right.real && imaginary == right.imaginary;

}

private:

int real;

int imaginary;

};

// overloaded << operator

ostream &operator<<( ostream &out, Complex &obj )

{

if(obj.imaginary>0)

out<<obj.real<<"+"<<obj.imaginary<<"i";

else if(obj.imaginary==0)

out<<obj.real;

else out<<obj.real<<"-"<<-obj.imaginary<<"i";

return out;

}

int main()

{

int a;

int b;

cout<<"Enter two integer values:";

cin>>a>>b;

cout<<a<<"and"<<b<<"are"<<(isEqualTo(a,b)? "equal":"not equal")<<'\n';

char c;

char d;

cout<<"\nEnter two character values:";

cin>>c>>d;

cout<<c<<"and"<<d<<"are"<<(isEqualTo(c,d)? "equal":"not equal")<<'\n';

double e;

double f;

cout<<"\nEnter two double values:";

cin>>e>>f;

cout<<e<<"and"<<f<<"are"<<(isEqualTo(e,f)? "equal":"not equal")<<'\n';

Complex g(10,5);

Complex h(10,5);

cout<<"\nThe class objects"<<g<<"and"<<h<<"are"<<(isEqualTo(g,h)? "equal":"not equal")<<'\n';

system("pause");

}