

Student Name _____ Section _____
 Instructor _____ Due Date _____

Project	1	2	3	4	5	6	TOTAL
Maximum Points	2 points	2 points	2 points	2 points	1 point	1 point	10 points
Your Score							

PROJECT ONE (Telephone Services - Telly Thomas)

Objective To write, compile and execute a program that generates a telephone bill.

PROJECT DESCRIPTION

Telly Thomas has two different services, cellular telephone and regular home telephone, with the same company, but receives a single telephone bill for both services.

For cellular telephone use, Telly is charged \$ 19.99 per month and \$ 0.75 per minute for either local or long distance calls. For regular home telephone use, Telly is charged \$ 13.29 per month, \$ 0.22 per minute for local calls and \$ 0.52 per minute for long distance calls.

A 6 % tax is added to the total monthly cellular telephone charge and an 11.5 % tax is added to the total monthly regular home telephone charge.

In addition to the above charges, Telly must pay the company a flat \$ 10 per month account maintenance fee, which is subject to a 3.0 % tax.

Note: prior month, past due amounts are accessed a 2.5 % penalty charge.

Write, compile and execute a program that computes the total monthly telephone bill for Telly Thomas. Test your program with the sample data given in **Figure 1**, below.

Figure 1 Telephone Use for the Month of June - Customer: Telly Thomas

	Regular Telephone	Cellular Telephone
Local Calls	79 minutes	63 minutes
Long Distance Calls	12 minutes	19 minutes

Amount Past Due \$ 12.50

Information About This Project

The main parts of this program are:

Input

Your program is to prompt the user to enter each of the following:

- the customer name
- the month of the billing
- the number of local cellular telephone calls
- the number of long distance cellular telephone calls
- the number of local regular telephone calls
- the number of long distance regular telephone calls
- the amount past due

Process

Your program is to compute each of the following:

- the total cellular charge (before taxes)
- the total regular telephone charge
- the amount past due (including the penalty)
- the total bill amount (including the total cellular charge, the total regular charge, the tax on these charges, the amount past due plus penalty and the account maintenance fee plus its associated tax)

Student Name _____

Section _____

PROJECT ONE**Output**

Your program is to output each of the following:

- the customer name
- the month of the billing
- the total cellular telephone charge (before taxes)
- the total regular telephone charge (before taxes)
- the amount past due (including penalty)
- the total bill amount

Steps To Complete This Project**STEP 1 Open a C++ Editor and Type the Program Code**

Open a C++ editor, such as MS Visual Studio. Type the program code given in **Figure 1**, which follows. Save your file as: Telephone.cpp

The program code consists of the desired input items and output items described above. The program is coded such that all dollar amounts will be displayed in a monetary format with a dollar sign and two decimal points.

In the heading portion of the code, include your name, the date, course number and instructor's name.

STEP 2 Compile and Run Your Program

After you type the program code, compile and run your program. If you have syntax, logic or run - time errors, debug the errors and recompile. Once you have determined that your program runs successfully, observe and verify the total bill amount. Perform a sample run with the information given on the prior page. Your output should appear as follows:

```
Enter the first and last name of the customer: Telly Thomas
Enter the billing month: June
Minutes of local cellular calls: 63
Minutes of long distance cellular calls: 19
Minutes of local regular telephone calls: 79
Minutes of long distance regular telephone calls: 12
Amount past due: 12.50

Monthly Telephone Bill for Telly Thomas
  Billing Month: June
  Charges (before tax):

    Total cellular charges: $81.49
    Total regular telephone charges: $36.91
    Total amount past due: $12.81
    Total Amount Charge: $150.65
```

STEP 3 Modify Your Program

With your program running correctly, modify it such that class Telephone includes a new member function named checkPastDute() which checks whether the amount past due exceeds \$ 100. If it does, a message indicating this is to be displayed to the user.

STEP 4 Submit Your Source Code

Submit hardcopy of your complete modified program source code for credit. Also submit screen snapshots of the operation of your application.

Student Name _____

Section _____

PROJECT ONE

Figure 1 Program Code for the Telephone Class

```
//Sammy Student Programmer

#include <iostream>
#include <string>
using namespace std;

class Telephone {

private:

    string firstName, lastName, month;
    float localCell, longCell, localReg, longReg;
    float pastDue;
    //class data members

public:

    Telephone() {
        firstName = "";
        lastName = "";
        month = "";
        localCell = 0;
        longCell = 0;
        localReg = 0;
        longReg = 0;
    } //class constructor

    string getName() {
        return firstName + " " + lastName;
    } //returns customer name

    string getMonth() {
        return month;
    } //returns the billing month

    float calcCell() {
        return (longCell + localCell) * 0.75 + 19.99;
    } //calculates the pretax cell charges

    float calcReg();
        //calculates pretax reg phone charges

    float getPast() {
        return pastDue * 1.025;
    } //calculates and returns the past due charges

    float calcTotal();
        //calculates and returns the total months charges

    void setValue();
        //prompts user to input
}; //end class definition
```

Student Name _____

Section _____

PROJECT ONE

Figure 1 Program Code for the Telephone Class (continued)

```

//class member functions
float Telephone::calcReg() {
    return (localReg * 0.22 + longReg * 0.52 + 13.29);
}

float Telephone::calcTotal() {
    float cellTax, regTax;
    float total;
    cellTax = calcCell() * 1.06;
    regTax = calcReg() * 1.115;
    total = 10 * 1.03 + cellTax + regTax + getPast();
    return total;
}

void Telephone::setValue()
{
    cout << "Enter the first and last name of the customer: ";
    cin >> firstName >> lastName;
    cout << "Enter the billing month: ";
    cin >> month;
    cout << "Minutes of local cellular calls: ";
    cin >> localCell;
    cout << "Minutes of long distance cellular calls: ";
    cin >> longCell;
    cout << "Minutes of local regular telephone calls: ";
    cin >> localReg;
    cout << "Minutes of long distance regular telephone calls: ";
    cin >> longReg;
    cout << "Amount past due: ";
    cin >> pastDue;
}

void main() {
    Telephone x;
    x.setValue();
    cout.setf(ios::fixed); //format output to show 2 decimal places
    cout.setf(ios::showpoint);
    cout.precision(2);
    cout << "\n\nMonthly Telephone Bill for " << x.getName();
    //output
    cout << "\n    Billing Month: " << x.getMonth();
    cout << "\n    Charges (before tax):\n";
    cout << "\n    Total cellular charges: $" << x.calcCell();
    cout << "\n    Total regular telephone charges: $" << x.calcReg();
    cout << "\n    Total amount past due: $" << x.getPast();
    cout << "\n    Total Amount Charge: $" << x.calcTotal();
    cout << "\n";
}

```

Student Name _____ Section _____

PROJECT TWO (Linear Depreciation)

Objective To type, compile and execute a program that uses a class to compute linear depreciation.

PROJECT DESCRIPTION

Natoma Telecommunications has purchased a business - use system costing \$ 5,000 . The pertinent information concerning this asset is listed in **Figure 1** below.

Natoma's accounting manager wants you to write a computer program, which will calculate the annual straight - line depreciation of this asset and generate a depreciation report based on the items in **Figure 1** as well as the annual depreciation amount.

Write the program using a class called **Linear**, which contains the variables cost, rate, time and depreciation along with a function to calculate linear depreciation.

Include a function to monitor the asset book value to ensure that it does not fall below the salvage value.

Figure 1 Business Asset Information

<i>Asset type</i>	Computer System
<i>Depreciation method</i>	Straight - Line
<i>Asset cost</i>	\$ 5,000.00
<i>Asset life</i>	5 years
<i>Salvage value</i>	\$ 500.00

Information About This Project

In accounting, the formula for straight - line depreciation is: $V(t) = C - CRt$

where $V(t)$ is the value of the asset at time t , C is the asset cost and R is the rate defined by:

$$R = \frac{\text{asset cost} - \text{salvage value}}{\text{asset cost} \times \text{asset life}}$$

The product CR represents the annual depreciation expense.

The book value of an asset is its asset cost subtracted by the accumulated depreciation.

Steps To Complete This Project**STEP 1 Open a C ++ Editor and Type the Program Code**

Open a C ++ editor on you computer. Type the program code given in **Figure 1**, which contains statements which accept the necessary input items and display the required output items for the above project.

Save your file as: `Linear.cpp`

Include your name, date and course title in the heading portion of your code.

STEP 2 Compile and Run your Program

Build, compile and run your program. Test your program.

Use the sample information above for the data for your sample run.

Your sample run should appear as follows:

Student Name _____

Section _____

PROJECT TWO

Asset Type: Computer
Asset Cost: 5000
Asset Life (years): 5
Salvage Value: 500

Computer Report:
Deprecation Rate: 18%
Annual Deprecation Expense: 900
Book Value: 4100

STEP 3 Modify Your Program

With your program running correctly, modify it such that all dollar amounts are displayed in a currency format with a dollar sign and two decimal places.

STEP 4 Submit Your Program Code

Submit your completed modified program code. Include for submission, a run time image of your program showing the results of running the sample information in **Figure 1**, with all dollar amounts displayed in a currency format.

Student Name _____

Section _____

PROJECT TWO

Figure 2 Program Code for the Linear Class

```
#include <iostream>
#include <string>
using namespace std;

class Linear {
private:
    float cost, rate, time, salvage;
    string type;
public:
    Linear() {
        cost = 0;
        rate = 0;
        time = 0;
        salvage = 0;
        type = "";
    } //constructor

    void setValue();
        //allows the user to input values for the asset

    string getType() { return type; }
        //returns the asset name/type
    double getRate() { return rate; }
        //returns the depreciation rate
    float getDeprecation() { return cost * rate; }
        //returns the annual depreciation expense
    float getBookValue() { return cost - getDeprecation(); }
        //returns the book value of the asset
    bool Monitor() { return (getBookValue() > salvage); }
        //compares the book value to the salvage value
        //returns 1 if book value > salvage, 0 if book value < salvage
};

void Linear::setValue() {
    cout << "Asset Type: ";
    cin >> type;
    cout << "Asset Cost: ";
    cin >> cost;
    cout << "Asset Life (years): ";
    cin >> time;
    cout << "Salvage Value: ";
    cin >> salvage;

    rate = (cost - salvage) / (cost * time);
}
```

Student Name _____

Section _____

PROJECT TWO**Figure 2 Program Code for the Linear Class (continued)**

```
void main()
{
    Linear asset;
    asset.setValue();
    cout << endl << endl << asset.getType() << " Report:\n";
    cout << "Deprecation Rate: " << asset.getRate() * 100
        << "%" << endl;
    cout << "Annual Deprecation Expense: " << asset.getDeprecation()
        << endl;
    cout << "Book Value: " << asset.getBookValue() << endl;
    if(asset.Monitor() == 0)
        cout << "Warning: Book value of " << asset.getType()
            << " has fallen below the salvage value." << endl;
}
```

Student Name _____ Section _____

PROJECT THREE (Class Rectangle)

Objective To type, compile and execute a program that uses a class to compute the area and perimeter of a rectangle.

PROJECT DESCRIPTION

Create a class entitled `Rectangle`. The class has attributes **length** and **width**, each of which defaults to the value of 1. The class has member functions that will calculate the perimeter and the area of the rectangle. It also has `set` and `get` functions for both **length** and **width**. The set functions should verify that the **length** and **width** the user enters are each floating point numbers larger than 0.0 and less than 20.00.

Information About This Project

Note: The *area* of a rectangle is: $\text{length} \times \text{width}$
The *perimeter* of a rectangle is: $2 \times \text{length} + 2 \times \text{width}$

Steps To Complete This Project**STEP 1 Open a C++ Editor and Type the Program Code**

Open a C++ editor on your computer. Write the program code necessary to accept the input items and display the required output items of the above project. Save your file as: `Rectangle.cpp`

Figure 1, which follows, contains an incomplete program which you can use as starter code for your program. Determine the missing entries to complete the starter code.

Include your name, date and course title in the heading portion of your code.

STEP 2 Compile and Run your Program

Build, compile and run your program. Test your program.

STEP 3 Print Your Program Code

Finally, print your program code and attach it to this lab packet for credit. Include for submission, a run time image of your program showing the results of running some sample information.

Student Name _____

Section _____

PROJECT THREE

Figure 1 Incomplete Program Code for the Rectangle Class

```

#include <iostream>
using namespace std;

class Rectangle {           //define class
public:
    _____() { length = 0; width = 0; } //constructor
    float _____() { return length; } //return length
    float getWidth() { return width; } //return width
    float area() { _____ length * width; } //return area
    float perimeter() { return 2 * length + 2 * width; }

    void setValues(float, float) ;

private:
    float length, width;
};

void _____::setValues(float len, float wid)
{
    if (len <= 0 || _____ >= 20) {
        cout << "Enter new value for length ";
        cin >> len;
    }
    length = len;

    if (wid <= 0 || wid >= 20) {
        cout << "Enter new value for width ";
        cin >> wid;
    }
    _____ = wid;
}

void main() {
    float length, width;
    Rectangle _____ ;

    cout << "Enter a value for length and width between 0 and 20\n";
    cin >> length >> width;

    r.setValues(length, width);

    cout << "\nGiven the length of " << r.getLength()
        << " and the width of "
        << r.getWidth() << " \nyour area is " << _____()
        << " and your perimeter is "
        << r.perimeter() << "\n\n";
}

```

Student Name _____ Section _____

PROJECT FOUR (Class Portfolio)

Objective To type, compile and execute a program that uses a class to manage an investment portfolio.

PROJECT DESCRIPTION

Create a class named **Portfolio** to manage a share portfolio. Share holdings are identified by the company name, the number of shares purchased, the date of the purchase, the cost per share and the current price per share. Your program must be able to calculate the value of the current shares, the profit or loss made on the stock purchase and the ability to sell the shares.

You can test your program with the sample data given in **Figure 1**, below. For the latest in stock market quotes, you can visit a Web site such as:

<http://www.nasdaq.com>

when the homepage of the **Nasdaq** site opens, type the particular stock symbols such as IBM within the Quotes grid to retrieve current stock market quotes.

Figure 1

Portfolio Information:	IBM stock
Number of Shares	100
Date of Purchase	04 -15 - 05
Cost per Share	\$ 82.00
Price per Share	\$ 84.50

Information About This Project

This program illustrates an example of using a class to define data members and use the members for computational purposes.

Steps To Complete This Project**STEP 1 Open a C++ Editor and Type the Program Code**

Open a C++ editor on your computer. Write the program code necessary to accept the input items and display the required output items of the above project. Save your file as: `Invest.cpp`

Include your name, date and course title in the heading portion of your code.

STEP 2 Compile and Run your Program

Build, compile and run your program. Test your program.

STEP 3 Submit Your Program Code

Finally, submit your program code for credit. Include for submission, a run time image of your program showing the results of running some sample information.

Student Name _____ Section _____

PROJECT FIVE (Iteration-The Trigonometric Sine Function)**Objective** To write, compile and execute a program that computes $\sin x$ by iteration.**PROJECT DESCRIPTION**

According to mathematics, the value of the trigonometric sine function of x , abbreviated as $\sin x$, can be approximated by summing the first n terms of the infinite series:

$$\sin x = x - x^3/3! + x^5/5! - x^7/7! + x^9/9! - x^{11}/11! + \dots$$

where x is expressed in radians, such that π radians = 180° , and where $1!$ represents 1 factorial, $2!$ represents 2 factorial, and so on.

Write, compile and execute a program that approximates the value of $\sin x$ for any user-defined real number x . Use a `do` loop or a `while` loop that terminates when the difference between any two successive approximations differs by less than $1.0 \text{ E} - 6$.

This means your program should be written such that the user is prompted to enter any value for x and the program then outputs the approximate value of $\sin x$.

Information About This Project

This program illustrates an example of using a series expansion to approximate the value of a trigonometric function.

Steps To Complete This Project**STEP 1 Open a C++ Editor and Type the Program Code**

Open a C++ editor on your computer. Write the program code necessary to accept the input items and display the required output items of the above project. Save your file as: `SineWave.cpp`

Include your name, date and course title in the heading portion of your code.

STEP 2 Compile and Run your Program

Build, compile and run your program. Test your program.

STEP 3 Submit Your Program Code

Submit your program code for credit. Include for submission, a run time image of your program showing the results of running some sample information.

STEP 4 Modify Your Program Code

Save your completed program.

Then, save the program as `CosineWave.cpp`. Then, modify the program to now approximate the value of $\cos x$ for any user-defined real number x . Use an Internet search engine to research and locate an infinite series for the cosine, similar to the one for the sine function shown above.

Hint: consider the sine as an odd function and the cosine as an even function.

STEP 5 Submit Your Modified Program Code

Submit your program code for credit. Include for submission, a run time image of your program showing the results of running some sample information.

Student Name _____

Section _____

PROJECT FIVE

Figure 1 Program Code for the Sine Approximation

```
#include <iostream>
#include <math.h>
using namespace std;

float approximate(float value, int p, float total);
//function that approximates the value of sin x by iteration
float calcTerm(float z, int m);
//calculates the mth term of the infinite series for sin x

void main() {
    float x;
    float y = 0;
    int n = 1;
    char repeat = 'Y';
    while(repeat == 'Y') {
        cout << "Enter a value for x in radians: ";
        cin >> x;

        cout << "Given a value x of " << x
            << " the approximate value of sin x is "
            << approximate(x, n, y);
        cout << endl << endl << "Run again? (Y/N) ";
        cin >> repeat;
        cout << endl;
    }
}

float approximate(float value, int p, float total) {
    float sum = 0;
    float diff;
    if(p % 4 == 3)
        sum = total - calcTerm(value, p);
    else if(p % 4 == 1)
        sum = total + calcTerm(value, p);
    diff = fabs(sum - total);
    if(diff > .000001)
        sum = approximate(value, p + 2, sum);
    return sum;
}

float calcTerm(float z, int m) {
    float term;
    term = z / m;
    while(m > 1) {
        m--;
        term = term * z / m;
    }
    return term;
}
```

Student Name _____ Section _____

PROJECT SIX (Management Science-Bin Packing)

Objective To write, compile and execute a program that solves a management science problem.

PROJECT DESCRIPTION

Shipping clerks at Sedgwick Specialty Services pack the company's top selling item (hand-held portable, personal communicators) in special containers. These containers are available in four sizes: giant, large, medium and small, which can hold 50, 20, 5 and 1 of these communicators, respectively.

Write a program that reads the number of communicators to be shipped and displays the number of giant, large, medium and small containers needed to send the shipment in the minimum number of containers, and with the minimal amount of unused space. Use constant definitions for the number of communicators each type of container can hold. The output of your program should be similar to the following.

container type	number of containers
-----	-----
Giant	21
Large	2
Medium	1
Small	3

Execute (Test) your program separately for 3, 19, 46, 75 and 1538 communicators.

Information About This Project

This program illustrates an example of management science and bin packing.

Steps To Complete This Project**STEP 1 Open a C++ Editor and Type the Program Code**

Open a C++ editor on your computer. Write the program code necessary to accept the input items and display the required output items of the above project. Save your file as: BinPack.cpp

Include your name, date and course title in the heading portion of your code.

STEP 2 Compile and Run your Program

Build, compile and run your program. Test your program.

STEP 3 Submit Your Program Code

Submit your program code for credit. Include, for submission, run time image(s) of your program showing the results of running your programming individually for 3, 19, 46, 75 and 1538 communicators.

STEP 4 Modify Your Program Code

Modify your program code such that it will include the total cost for each individual container type. Assume each of the communicators cost \$ 5.00 each. A sample output for your modified program follows.

Student Name _____

Section _____

PROJECT SIX

Sedgwick Specialty Services

Number of Communicators to Ship: 1098

container type	number of containers	total cost per container type
----------------	----------------------	-------------------------------

Giant	21	\$ 5,250
Large	2	\$ 200
Medium	1	\$ 25
Small	3	\$ 15

STEP 5**Submit Your Modified Program Code**

Submit your modified program code for credit. Include, for submission, run time image(s) of your program showing the results of running your program for 2,115 communicators.