EGR 125 Name: \_David Vermaak\_

Introduction to Engineering Methods (C++) Due Date: \_\_9/23/21\_\_\_

File: N125-Ch4LB

**Chapter 4 Homework**

**Reading Assignment:**

Read Chapter 4 in Introduction to Programming with C++, 3rd Edition, by Liang

**Problem Assignment:**

Submit each of the following by the assigned due date.

1. (40 pts) Work the following problems in the textbook. Work each problem by hand (not using the compiler). ***Be sure to write out the instructions for each problem and include the given information.*** Work all parts for each problem unless otherwise noted.

**Checkpoint Exercises:**

* **4.1**: Assume PI is 3.14159 and E is 2.71828, evaluate the following function calls:

|  |  |  |  |
| --- | --- | --- | --- |
| (a) sqrt(4.0) | *2* | (j) floor(-2.5) | -3 |
| (b) sin(2 \* PI) | *-5.07036e-006* | (k) asin(0.5) | 0.523599 |
| (c) cos(2 \* PI) | *1* | (l) acos(0.5) | 1.0472 |
| (d) pow(2.0, 2) | *4* | (m) atan(1.0) | 0.785398 |
| (e) log(E) | *0.999999* | (n) ceil(2.5) | 3 |
| (f) exp(1.0) | *2.71828* | (o) floor(2.5) | 2 |
| (g) max(2, min(3, 4)) | *3* | (p) log10(10.0) | 1 |
| (h) sqrt(125.0) | *11.1803* | (q) pow(2.0, 3) | 8 |
| (i) ceil(-2.5) | *-2* |  |  |

* **4.8**: Show the printout of the following code:

int i = '1';

int j = '1' + '2';

int k = 'a';

char c = 90;

cout << i << " " << j << " " << k << " " << c << endl;

\_\_\_\_\_\_\_\_\_\_\_\_\_

***49 99 97 Z***

\_\_\_\_\_\_\_\_\_\_\_\_\_

* **4.9**: Show the printout of the following code:

char c = 'A';

int i = c;

float f = 1000.34f;

int j = f;

double d = 1000.34;

int k = d;

int l = 97;

char ch = l;

cout << c << endl;

cout << i << endl;

cout << f << endl;

cout << j << endl;

cout << d << endl;

cout << k << endl;

cout << l << endl;

cout << ch << endl;

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

***A***

***65***

***1000.34***

***1000***

***1000.34***

***1000***

***97***

***a***

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* **4.10:** Show the output of the following program:

#include <iostream>

using namespace std;

int main()

{

char x = 'a';

char y = 'c';

cout << ++x << endl;

cout << y++ << endl;

cout << (x - y) << endl;

return 0;

}

\_\_\_\_\_\_\_\_

***b***

***c***

***-2***

**\_\_\_\_\_\_\_\_**

* **4.20:** Show the output of the following code:

string s1 = "Good morning";

string s2 = "Good afternoon";

cout << s1[0] << endl;

cout << (s1 == s2 ? "true": "false") << endl;

cout << (s1 != s2 ? "true": "false") << endl;

cout << (s1 > s2 ? "true": "false") << endl;

cout << (s1 >= s2 ? "true": "false") << endl;

cout << (s1 < s2 ? "true": "false") << endl;

cout << (s1 <= s2 ? "true": "false") << endl;

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

***G***

***false***

***true***

***true***

***true***

***false***

***false***

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* **4.23:** Show the output of the following statements.

cout << setw(10) << "C++" << setw(6) << 101 << endl;

cout << setw(8) << "Java" << setw(5) << 101 << endl;

cout << setw(6) << "HTML" << setw(4) << 101 << endl;

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

***C++ 101***

***Java 101***

***HTML 101***

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* **4.24:** Show the output of the following statements:

double number = 93123.1234567;

cout << setw(10) << setprecision(5) << number;

cout << setw(10) << setprecision(4) << number;

cout << setw(10) << setprecision(3) << number;

cout << setw(10) << setprecision(8) << number;

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

***931239.312e+004 9.31e+004 93123.123***

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

* **4.27:** Show the output of the following statements.

cout << left;

cout << setw(6) << 21.23 << endl;

cout << setw(6) << 51.34 << endl;

\_\_\_\_\_\_\_\_\_\_\_\_\_

***21.23***

***51.34***

\_\_\_\_\_\_\_\_\_\_\_\_\_

2) (60 pts) **Create a flowchart** and write C++ programs for each of the following. For each program:

* Use the TCC template (see next page). Complete the items listed in the template.
* Include plenty of comments.
* The output should be neatly and clearly formatted.
* Use good programming style and proper indentation.
* Turn in a printout of the program and printouts for all required test cases.

A) ***Letter Grade to Number:*** Write a program for ***Programming Exercise 4.12*** (Convert letter grade to number) Write a program that prompts the user to enter a letter grade A/a, B/b, C/c, D/d, or F/f and displays its corresponding numeric value 4, 3, 2, 1, or 0.

Run the program for:

* + Two valid lower case grades
  + Two valid upper case grades
  + Two invalid characters

B) ***Check SSN***: Write a program for ***Programming Exercise 4.23*** (Check SSN) Write a program that prompts the user to enter a Social Security number in the format ddd-dd-dddd, where d is a digit. Additional notes or specifications:

* Read the input as a single string.
* Check the string to be sure that it is of the correct length (11). Display an appropriate error message if the string is not the correct length.
* Check each symbol in the string for the appropriate ASCII value or range. Display an appropriate error message if the string contains any invalid symbols.
* If the SSN is valid display an appropriate message.
* Testing: Run the program for:

123-45-6789

263-48-3486

An input with the incorrect length

An input with an incorrect symbol.

C) ***Formatted Table***: Write a program to print a small formatted table showing the radius, diameter, circumference, and area of two circles according to the following specifications:

* Prompt the user to enter the radius of two circles (in cm).
* Display 3 tables. Each table should consist of the table number, the heading, and the values of radius, diameter, circumference, and area using the specified formatting.
  + Table 1: Use the following formatting:
    - Use the table heading shown in the sample output
    - Left justification
    - Fixed format
    - 3 digits after the decimal point
    - Show trailing zeros
  + Table 2: Use the following formatting:
    - Use the table heading shown in the sample output
    - Right justification
    - Fixed format
    - 5 digits after the decimal point
    - Show trailing zeros
  + Table 3: Use the following formatting:
    - Use the table heading shown in the sample output
    - Left justification
    - Scientific format
    - 2 digits after the decimal point
    - Show trailing zeros

***Sample Output***

Enter the radius for circle #1 (in cm): 2

Enter the radius for circle #2 (in cm): 5

Table #1:

Radius(cm) Diameter(cm) Circumference(cm) Area(cm^2)

2.000 4.000 12.566 12.566

5.000 10.000 31.416 78.540

Table #2:

Radius(cm) Diameter(cm) Circumference(cm) Area(cm^2)

2.00000 4.00000 12.56637 12.56637

5.00000 10.00000 31.41593 78.53982

Table #3:

Radius(cm) Diameter(cm) Circumference(cm) Area(cm^2)

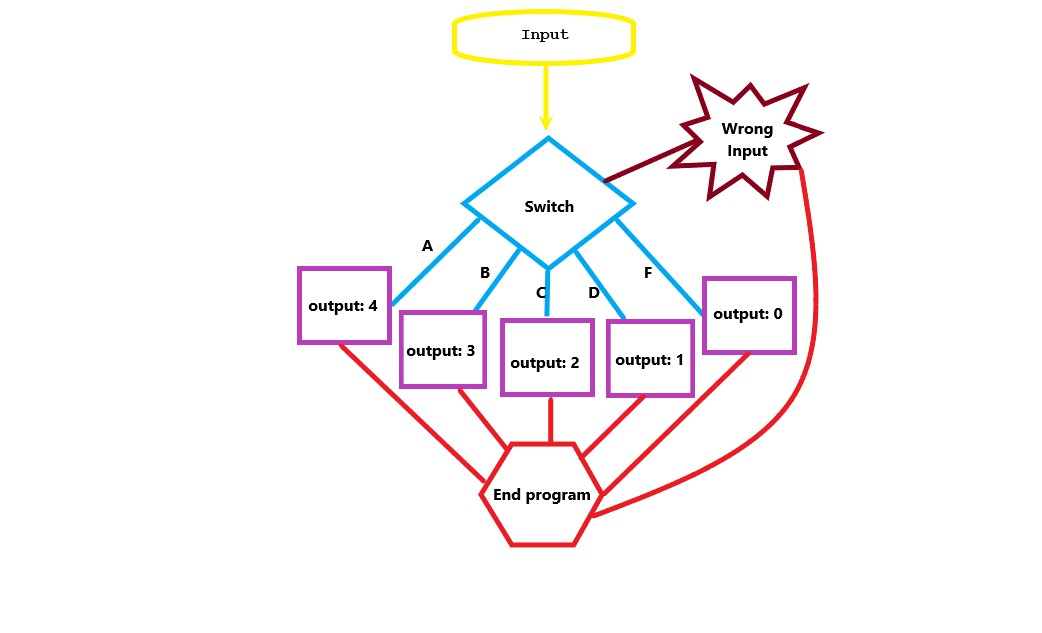
2.00e+000 4.00e+000 1.26e+001 1.26e+001

5.00e+000 1.00e+001 3.14e+001 7.85e+001

* Testing: Run the program for:
  + The example above (radius values of 2 and 5)
  + Another example using radius values between 0.1 and 1
  + Another example using radius values between 10 and 100

Note: If you copy the output into Word or Notepad, you might notice that your tables are no longer properly aligned. Change the font to ***Courier New*** (not a TrueType font) to fix the problem.

**PROGRAM A Flowchart:**

****

**PROGRAM A: Letter Grade to Number**

// \_\_\_\_ \_\_ \_\_

// / / \ / \

// / / /

// / \\_\_\_/ \\_\_\_/

//

// Date: 9/21/2021

// Name: David Vermaak

// Project: Letter grade to number

// Inputs: char

// Outputs: int

// Program Description: this program calculates the numeric value of letter grades

//

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

#include <iostream> //header containing cout and cin

using namespace std; //introduces namespace std needed to use cout and cin

int main ()

{

char grade; //initializes a variable character called grade

cout << "Please enter a letter grade (A, B, C, D, or F)\n"; //asks user for input

cin >> grade;

switch(grade) //uses a switch to jump to the correct line

{

case 'A': case 'a':

cout << "4\n";

break;

case 'B': case 'b':

cout << "3\n";

break;

case 'C': case 'c':

cout << "2\n";

break;

case 'D': case 'd':

cout << "1\n";

break;

case 'F': case 'f':

cout << "0\n";

break;

default:

cout << "Invalid Selection. Try Again\n"; //error message for incorrect

// input

}

return 0;

}

**OUTPUT FOR PROGRAM A:**

**A.1:**

*Please enter a letter grade (A, B, C, D, or F)*

*b*

*3*

**A.2:**

*Please enter a letter grade (A, B, C, D, or F)*

*c*

*2*

**A.3:**

*Please enter a letter grade (A, B, C, D, or F)*

*A*

*4*

**A.4:**

*Please enter a letter grade (A, B, C, D, or F)*

*F*

*0*

**A.5:**

*Please enter a letter grade (A, B, C, D, or F)*

*q*

*Invalid Selection. Try Again*

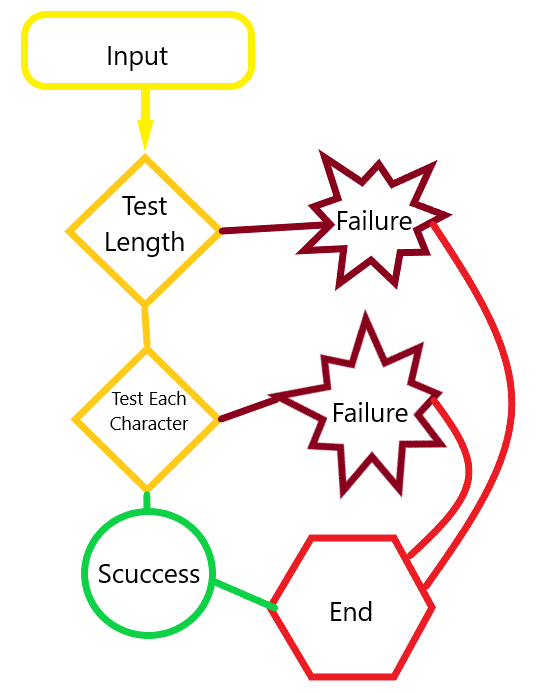
**A.6:**

*Please enter a letter grade (A, B, C, D, or F)*

*6*

*Invalid Selection. Try Again*

**PROGRAM B Flowchart:**

**

**PROGRAM B: Check SSN**

// \_\_\_\_ \_\_ \_\_

// / / \ / \

// / / /

// / \\_\_\_/ \\_\_\_/

//

// Date: 9/21/2021

// Name: David Vermaak

// Project: Checking the validity of an SSN

// Inputs: string

// Outputs: text

// Program Description: this program checks the validity of an SSN

//

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

#include <iostream> //header containing cout and cin

using namespace std; //introduces namespace std needed to use cout and cin

int main ()

{

string SSN;

bool t1,n1,n2,n3,n4,n5,n6,n7,n8,n9,d1,d2;

cout << "Please enter your Social Security Number in ddd-dd-dddd format where d is a digit\n";

cin >> SSN;

//Using bools to test if each char of the string is the correct type

n1 = (isdigit(SSN[0] ));

n2 = (isdigit(SSN[1] ));

n3 = (isdigit(SSN[2] ));

d1 = ((SSN[3]) == '-' );

n4 = (isdigit(SSN[4] ));

n5 = (isdigit(SSN[5] ));

d2 = ((SSN[6]) == '-' );

n6 = (isdigit(SSN[7] ));

n7 = (isdigit(SSN[8] ));

n8 = (isdigit(SSN[9] ));

n9 = (isdigit(SSN[10] ));

//by multiplying bools you can check if one or more are false because 0\*n = 0

t1 = n1\*n2\*n3\*n4\*n5\*n6\*n7\*n8\*n9\*d1\*d2;

if (11 == (SSN.length()))

{

if (t1 == 1)

{

cout << endl << SSN << " is a valid SSN\n" ;

}

else

cout << "You have used an incorrect character, do not use spaces, use the - symbol\n";

return 0;

}

else

cout << "Sorry, your SSN is not the correct length \n";

return 0;

}

**OUTPUT FOR PROGRAM B:**

**B.1:**

*Please enter your Social Security Number in ddd-dd-dddd format where d is a digit*

*123-45-6789*

*123-45-6789 is a valid SSN*

**B.2:**

*Please enter your Social Security Number in ddd-dd-dddd format where d is a digit*

*263-48-3486*

*263-48-3486 is a valid SSN*

**B.3:**

*Please enter your Social Security Number in ddd-dd-dddd format where d is a digit*

*123-456 68790*

*Sorry, your SSN is not the correct length (do not use spaces, use the - symbol)*

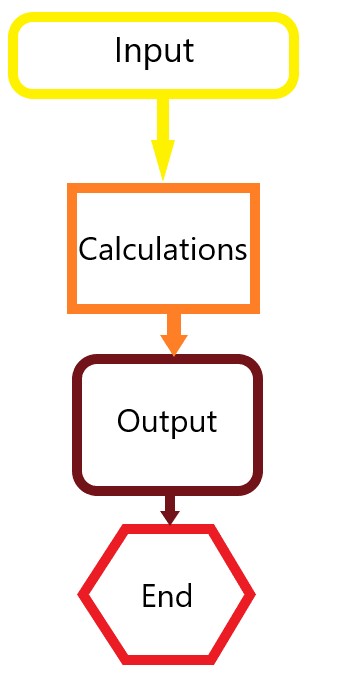
**B.4:**

*Please enter your Social Security Number in ddd-dd-dddd format where d is a digit*

*123\_91\_6789*

*You have used an incorrect character (do not use spaces, use the - symbol)*

**PROGRAM C Flowchart:**

**

**PROGRAM C: Formatted Table**

// \_\_\_\_ \_\_ \_\_

// / / \ / \

// / / /

// / \\_\_\_/ \\_\_\_/

//

// Date: 9/21/2021

// Name: David Vermaak

// Project: Formatted table

// Inputs: string

// Outputs: text

// Program Description: this program

//

//\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

#include <iostream> //header containing cout and cin

using namespace std; //introduces namespace std needed to use cout and cin

int main ()

{

const float pi = 3.14159265; //sets pi as a constant

float r1, r2, d1, d2, c1, c2, a1, a2;

cout << left << "Enter the radius for circle #1 (in cm): ";

cin >> r1;

cout << endl << "Enter the radius for circle #2 (in cm): ";

cin >> r2;

//all the math calculations

d1 = r1 \* 2;

d2 = r2 \* 2;

c1 = 2 \* pi \* r1;

c2 = 2 \* pi \* r2;

a1 = pi \* pow(r1,2);

a2 = pi \* pow(r2,2);

cout << endl << left << "\nTable #1:\n"; //Table one

cout <<"Radius(cm) \t" << setw(16) << "Diameter(cm)" << setw(22) << "Circumference(cm)" << setw(16) <<"Area(cm^2)" << endl;

cout << fixed << setprecision(3) << setw(16) << r1 << setw(16) << d1 << setw(22) << c1 << setw(16) << a1 << endl;

cout << fixed << setprecision(3) << setw(16) << r2 << setw(16) << d2 << setw(22) << c2 << setw(16) << a2 << endl;

cout << endl << right << "Table #2:\n"; //Table 2

cout << setw(16) << right <<"Radius(cm)" << setw(16) << right << "Diameter(cm)" << setw(22) << right << "Circumference(cm)" << setw(16) << right <<"Area(cm^2)" << endl;

cout << fixed << setprecision(5) << setw(16) << right << r1 << setw(16) << right << d1 << setw(22) << right << c1 << setw(16) << right << a1 << endl;

cout << fixed << setprecision(5) << setw(16) << right << r2 << setw(16) << right << d2 << setw(22) << right << c2 << setw(16) << right << a2 << endl;

cout << endl << left << "\nTable #3:\n"; //Table 3

cout <<"Radius(cm) \t" << setw(16) << "Diameter(cm)" << setw(22) << "Circumference(cm)" << setw(16) <<"Area(cm^2)" << endl;

cout << scientific << showpoint << setprecision(2) << setw(16) << r1 << setw(16) << d1 << setw(22) << c1 << setw(16) << a1 << endl;

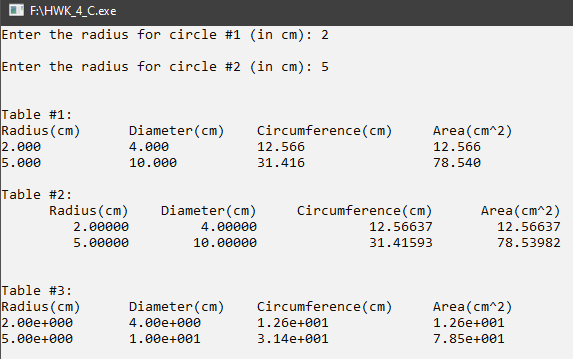
cout << scientific << showpoint << setprecision(2) << setw(16) << r2 << setw(16) << d2 << setw(22) << c2 << setw(16) << a2 << endl;

return 0;

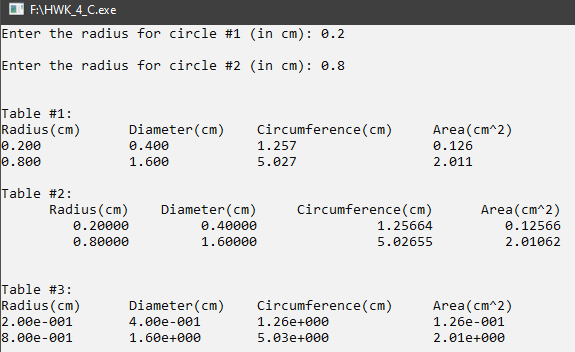
}

**OUTPUT FOR PROGRAM C:**

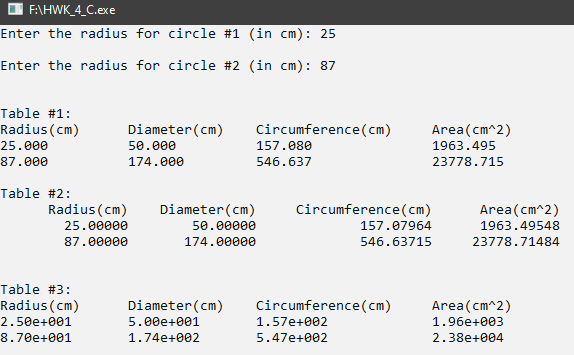
**C.1**

****

**C.2**

****

**C.3**

****