Problems with Statements

Problem 1(Sequential Statement)

a) Write a program, which asks the user to enter two numbers, and then output them after swapping their values using **third** variable.
b) Write a program, which asks the user to enter two numbers, and then output them after swapping their values without using third variable.

   Hint: Use arithmetic operators such as division '/' and multiplication * or addition + and subtraction

Problem 2(Sequential Statement)

Write a C++ program that take inputs a, b, c, d, e, f, g and h from user and calculate the result from the following mathematical equation:

$$\frac{1+2a}{3}+\frac{4(b+c)(5-d-e)}{f}-6\left(\frac{7}{g}+h\right)$$

Use proper data types.

Problem 3(Control Selection)

Write a program in C++ to read any day number in integer and display day name in the word.

Problem 4(Control Selection)

Write a program that input user choice to get character to ASCII or ASCII to character value. If 1 is pressed then program input character from user and show its ASCII value and if 2 is pressed then get input ASCII value and show its character.

Problem 5(Control Selection)

Write a c++ program to accept a coordinate point in a XY coordinate system and determine in which quadrant the coordinate point lies.

1. If both x and y are positive, then display point lies in first quadrant
2. If both x and y are negative, then display point lies in third quadrant
3. If both x is positive and y is negative, then display point lies in fourth quadrant.
4. If both x is negative and y is positive, then display point lies in second quadrant.
5. If both x and y are zero, then display point lies on origin
6. If x is zero and y is negative or positive, then display point lies exactly on line.
7. If y is zero and x is negative or positive, then display point lies exactly on line.

Problem No 6(Control Selection)

Write a program to perform the basic calculator operations using switch statement Demo of your program should just like that: First user will enter two variables and then program ask for the operation to be performed to those two variables.

```
enter 1st Number

10

enter 2st Number

5
Enter operator i.e. +,-,*,/
+
Sum of 1st and 2nd number is :15

Press any key to continue . . .
```

Problem 7(Loop)

Write an algorithm and make flow chart for finding if given three digit number is Armstrong number or not.

Definition of Armstrong number: An Armstrong number of three digits is an integer such that the sum of the cubes of its digits is equal to the number itself.

In other word "A number is Armstrong if it is equal to the sum of cube of its digits."

Example of Armstrong number is 371 as explained below
Armstrong number 371=(3)3+(7)3+(1)3
371=27+343+1
371=371

Problem No 8(Loop)

Input a number from user and will compute and display the factorial of that number.

Enter number= 4

Factorial = 4x3x2x1=24

Problem No 9(Loop)

Recall that in C++, while loops are used when a certain statement(s) must be executed repeatedly until certain conditions are met. Following is a C++ program that uses a while loop to find a Fibonacci number.

Consider the following sequence of numbers:

1, 1, 2, 3, 5, 8, 13, 21, 34, ...

Given the first two numbers of the sequence (say, a1 and a2), the nth number an, n >=3, of this sequence is given by:

An= an-1 + an-2

Thus:

a3 = a2 + a1 = 1 + 1 = 2;

a4 = a3 + a2 = 2 + 1 = 3;

and so on.

Such a sequence is called a Fibonacci sequence. In the preceding sequence, a2 = 1 and a1 = 1. However, given any first two numbers, using this process, you can determine the nth number, an,n >= 3, of the sequence. The number determined this way is called the nth Fibonacci number. Suppose a2 = 6 and a1 = 3.

Then:

a3 = a2 + a1 = 6 + 3 = 9; a4 = a3 + a2 = 9 + 6 = 15

Next, we write a program that determines the **nth** Fibonacci number given the first two numbers.

Input the **first two Fibonacci numbers of user choice** and the **nth(limit) number**.

Output The nth Fibonacci number.

Problem No10(Loop)

Write a program that presents the user choice of 5 favorite beverages (Coke, Fanta, Sprite, Pepsi, mineral water) then allow the user to choose a beverage by entering a number 1-5. Output which beverage they chose. User can give choices until he/she has money in his/her account and display message "out of money" with remaining balance when user has not enough money to buy a minimum cost drink (You must use switch statement).

Note:

1. Price of each drink is different and greater than zero (You can initialize prices).
2. Input Balance from user.
3. Input balance must be greater than minimum price drink.
4. Each time user enters his/her choice either he/she want to buy drink or not (use bool data type).
5. Display number of drinks with remaining balance if user did not want to buy drink or out of balance.

Problem No 11(Loop)

Write a program that input a number from user and check it is palindrome or not. If the reversed integer is equal to the integer entered by user then that number is a palindrome if not that number is not a palindrome. Your entered number must be non-zero

Problem No12(Loop)

The state of Florida maintains a census file in which each record contains the name of a county, the current population, and a number representing the rate at which the population is increasing per year. For example, one record might contain Miami-Dade County, 2,253,000, and 2 percent. The governor wants a report that lists each county and the number of years it will take for the population of the county to double, assuming the present rate of growth remains constant. Design an application that reads records from user and displays the county's name and the number of years it will take for the population to double. If a county's record contains a negative growth rate, then instead of displaying the number of years it takes for the population to double, display a message that indicates

that the population is never expected to double.

Problem No13(Loop)

When you borrow money to buy a house, a car, or for some other purpose, you repay the loan by making periodic payments over a certain period of time. Of course, the lending company will charge interest on the loan. Every periodic payment consists of the interest on the loan and the payment toward the principal amount. To be specific, suppose that you borrow $1000 at the interest rate of 7.2% per year and the payments are monthly. Suppose that your monthly payment is $25. Now, the interest is 7.2% per year and the payments are monthly, so the interest rate per month is 7.2/12 = 0.6%. The first month's interest on $1000 is 1000 * 0.006 = 6. Because the payment is $25 and interest for the first month is $6, the payment toward the principal amount is 25 – 6 = 19. This means after making the first payment, the loan amount is 1000 – 19 = 981. For the second payment, the interest is calculated on $981.

So, the interest for the second month is 981 * 0.006 = 5.886, that is, approximately $5.89. This implies that the payment toward the principal is 25 – 5.89 = 19.11 and the remaining balance after the second payment is 981 – 19.11 = 961.89. This process is repeated until the loan is paid.

Write a program that accepts as input the loan amount, the interest rate per year, and the monthly payment. (Enter the interest rate as a percentage. For example, if the interest rate is 7.2% per year, then enter 7.2.) The program then outputs the number of months it would take to repay the loan. (Note that if the monthly payment is less than the first month's interest, then after each payment, the loan amount will increase. In this case, the program must warn the borrower that the monthly payment is too low, and with this monthly payment, the loan amount could not be repaid.)

Problem No14(Pre/Post Increment & Decrement Operators)

Run the program below, comment on each line properly with functionality of that command in main(). Attach output of program also.

```
#include <iostream>

using namespace std;

int main()

{

	int x = 1, a;

	a = x++;

	cout << "\na = " << a;

	x++;

	a = x;

	cout <<"\na = " << a;

	a = ++x;
```

```
        cout << "\na = " << a;

        ++x;

        a = x;

        cout << "\na = " << a;

        system("pause");

        return 0;

}
```

Problems with errors

Problem No 1(Syntax error)

The following program contains errors. Correct them so that the program will run and also mention reason of error if any.

```
#include <iostream>

using namespace std;

const int SECRET = 5

main ()
{
    int x, y, w, z;
    z = 9;

    if z > 10
        x = 12; y = 5, w = x + y + SECRET;
    else
        x = 12; y = 4, w = x + y + SECRET;

    cout << "w = " << w << endl;
}
```

Problem No 2(Logical error)

State whether the following are valid switch statements. If not, explain why. Assume that n and digit are int variables.

a.
```
switch (n <= 2)
{
case 0:
    cout << "Draw." << endl;
    break;
case 1:
    cout << "Win." << endl;
    break;
case 2:
    cout << "Lose." << endl;
    break;
}
```

b.
```
switch (digit / 4)
{
case 0,
case 1:
    cout << "low." << endl;
    break;
case 1,
case 2:
    cout << "middle." << endl;
    break;
case 3:
    cout << "high." << endl;
}
```

c.
```
switch (n % 6)
{
case 1:
case 2:
case 3:
```

```
case 4:
case 5:
    cout << n;
    break;
case 0:
    cout << endl;
    break;
}
```

d.
```
switch (n % 10)
{
case 2:
case 4:
case 6:
case 8:
    cout << "Even";
    break;
case 1:
case 3:
case 5:
case 7:
    cout << "Odd";
    break;
}
```

Problem No 3()

In the following code, correct any errors that would prevent the program from compiling or running.

```
include <iostream>

main ()
{
    int a, b;
    bool found;
    cout << "Enter two integers: ;
    cin >> a >> b;

    if  a > a*b  &&  10 < b
        found = 2 * a > b;
    else
    {
        found = 2 * a < b;
        if found
            a = 3;
            c = 15;
            if b



            {
                b = 0;
                a = 1;
            }
}
```

Problem No 4()

The statements in the following program are in incorrect order. Rearrange the statements so that they prompt the user to input the shape type (rectangle, circle, or cylinder) and the appropriate dimension of the shape. The program then outputs the following information about the shape: For a rectangle, it outputs the area and perimeter; for a circle, it outputs the area and circumference; and for a cylinder, it outputs the volume and surface area. After rearranging the statements, you program should be properly indented.

```
using namespace std;

#include <iostream>

int main()
{
    string shape;
    double height;

    #include <string>

    cout << "Enter the shape type: (rectangle, circle, cylinder) ";
    cin >> shape;
    cout << endl;

    if (shape == "rectangle")
    {
        cout << "Area of the circle = "
             << PI * pow(radius, 2.0) << endl;

        cout << "Circumference of the circle: "
             << 2 * PI * pow(radius, 2.0) << endl;

        cout << "Enter the height of the cylinder: ";
        cin >> height;
        cout << endl;

        cout << "Enter the width of the rectangle: ";
        cin >> width;
        cout << endl;
```

```
        cout << "Perimeter of the rectangle = "
             << 2 * (length + width) << endl;
        double width;
    }

    cout << "Surface area of the cylinder: "
         << 2 * radius *  + 2 * PI * pow(radius, 2.0) << endl;
    }
    else if (shape == "circle")
    {
        cout << "Enter the radius of the circle: ";
        cin >> radius;
        cout << endl;

        cout << "Volume of the cylinder = "
             << PI * pow(radius, 2.0)* height << endl;
        double length;
    }
    return 0;
    else if (shape == "cylinder")
    {
        double radius;

        cout << "Enter the length of the rectangle: ";
        cin >> length;
        cout << endl;

        #include <iomanip>


        cout << "Enter the radius of the base of the cylinder: ";
        cin >> radius;
        cout << endl;

        const double PI = 3.1416;
        cout << "Area of the rectangle = "
             << length * width << endl;
    else
        cout << "The program does not handle " << shape << endl;
        cout << fixed << showpoint << setprecision(2);

    #include <cmath>
}
```

Problem No 5(Loop)

The following program has more than five mistakes that prevent it from compiling and/or running. Correct all such mistakes.

```
#include <iostream>

using namespace std;
const int N = 2,137;

main ()
{
    int a, b, c, d:

    a := 3;
    b = 5;
    c = c + d;
    N = a + n;
    for (i = 3; i <= N; i++)
    {
        cout << setw(5) << i;
        i = i + 1;
    }
    return 0;
}
```

Problems with dry runs

Problem No 1(Control Selection)

What is the output of the following program?

```
#include <iostream>

using namespace std;

int main()
{
    int myNum = 10;
    int yourNum = 30;

    if (yourNum % myNum == 3)
    {
        yourNum = 3;
        myNum = 1;
    }
    else if (yourNum % myNum == 2)
    {
        yourNum = 2;
        myNum = 2;
    }
    else
    {
        yourNum = 1;
        myNum = 3;
    }

    cout << myNum << " " << yourNum << endl;

    return 0;
}
```

a) What is the output of the program, if **myNum = 5** and **yourNum = 12**?
b) What is the output of the program, if **myNum = 30** and **yourNum = 33**?

Problem No 2(Control Selection)

Suppose the input is 3, 5 and 10. What is the value of beta after the following C++ code executes?

```
cin >> beta;
switch (beta)
{
case 3:
    beta = beta + 3;
```

```
case 1:
    beta++;
    break;
case 5:
    beta = beta + 5;
case 4:
    beta = beta + 4;
}
```

Problem No 3(loop)

Consider the following two loops

a.
```
i = 11;
while (i <= 10)
{
    cout << i << " ";
    i = i + 5;
}
cout << endl;
```

b.
```
i = 11;
do
{
    cout << i << " ";
    i = i + 5;
}
while (i <= 10);

cout << endl;
```

Show output of both loops.

Problem No 4(loop)

What is the output of the following program?

```
#include <iostream>

using namespace std;

int main()
{
    int  x, y, z;

    x = 4;   y = 5;
    z = y + 6;

    while(((z - x) % 4) != 0)
    {
        cout << z << " ";
        z = z + 7;
    }



     cout << endl;

     return 0;
 }
```

Problem No 5(Loop)

The following program is designed to input two numbers and output their sum. It asks the user if he/she would like to run the program. If the answer is Y or y, it prompts the user to enter two numbers. After adding the numbers and displaying the results, it again asks the user if he/she would like to add more numbers. However, the program fails to do so. Correct the program so that it works properly.

```
#include <iostream>
#include <iomanip>

using namespace std;

int main()
{
    char response;
    double num1;
    double num2;

    cout << "This program adds two numbers." << endl;
    cout << "Would you like to run the program: (Y/y) ";
    cin >> response;
    cout << endl;
```

```
    cout << fixed << showpoint << setprecision(2);

    while (response == 'Y' && response == 'y')
    {
        cout << "Enter two numbers: ";
        cin >> num1 >> num2;
        cout << endl;

        cout << num1 << " + " << num2 << " = " << (num1 - num2)
             << endl;

        cout << "Would you like to add again: (Y/y) ";
        cin >> response;
        cout << endl;
    }

    return 0;
}
```

Problem No 6(loop)

The do...while loop in the following program is supposed to read some numbers until it reaches a sentinel (in this case, -1). It is supposed to add all of the numbers except for the sentinel. If the data looks like: **12 5 30 48 -1** the program does not add the numbers correctly. Correct the program so that it adds the numbers correctly.

```
#include <iostream>

using namespace std;
int main()
{
   int total = 0,
       count = 0,
       number;
   do
   {
       cin >> number;
       total = total + number;
       count++;
   }
   while (number != -1);

   cout << "The number of data read is " << count << endl;
   cout << "The sum of the numbers entered is  " << total
        << endl;

   return 0;
}
```

Problem No 7(loop)

What is the output of the following code?

```
int num = 12;

while (num >= 0)
{
    if (num % 5 == 0)
        break;

    cout << num << " ";
    num = num - 2;
}

cout << endl;
```

What does a **break** statement do in a loop?

Problem No 8(loop)

What is the output of the following code?

```
int num = 12;

while (num >= 0)
{
    if (num % 5 == 0)
    {
        num++;
        continue;
    }

    cout << num << " ";
    num = num - 2;
}
cout << endl;
```

What does a **continue** statement do in a loop?