

CMPT 130: Lab Work Week 3

1. Multiple if versus if-else if-else Versus statements

What is the output of the following two C++ code fragments? What is the difference between the following two code fragments?

Program1.cpp

```
int main()
{
    int a = 5, b = 3;
    if (a == b)
        cout << a << " is equal to " << b << endl;
    if (a != b)
        cout << a << " is not equal to " << b << endl;
    if (a < b)
        cout << a << " is less than " << b << endl;
    if (a > b)
        cout << a << " is greater than " << b << endl;
    system("Pause");
    return 0;
}
```

Program2.cpp

```
int main()
{
    int a = 5, b = 3;
    if (a == b)
        cout << a << " is equal to " << b << endl;
    else if (a != b)
        cout << a << " is not equal to " << b << endl;
    else if (a < b)
        cout << a << " is less than " << b << endl;
    else
        cout << a << " is greater than " << b << endl;
    system("Pause");
    return 0;
}
```

Remark:

- If more than one condition evaluates to True then the multiple if statements is different from if - else if - else statements. If only one condition evaluates to true then they are equivalent; although the later is more efficient.
- Try this by making both a and b to have the same values and you will get the same output from Program1.cpp and Program2.cpp

2. Write a C++ program that declares six variables named y1, m1, d1, y2, m2, and d2 all as integer data types. Now read the birth date of a child1 in the y1, m1 and d1 variables where y1, m1 and d1 represent the year, month and day of the birth date of child1. Then

read the birth date of child2 in y2, m2, and d2 variables. Finally print the number of years, number of months and number of days between the birth dates of the two children.

Assume each month has 30 days and each year has 12 months (= 360 days).

Please note that your program should **NEVER** print any negative years or months or days between the birth dates of the two children. Instead the printed values of years, months and days between the birth dates of the two children **MUST** be positive or zero values.

For example, d1 = 18, m1 = 4, y1 = 1998 and d2 = 10, m2 = 9, y2 = 1995 then your program must print

The difference between the two children's birth dates is 2 years, 7 months and 8 days.

As another example, d1 = 10, m1 = 9, y1 = 1995 and d2 = 18, m2 = 4, y2 = 1998 then your program must still print

The difference between the two children's birth dates is 2 years, 7 months and 8 days.

3. Write a C++ program that declares three float inputs (call them a, b, c) that represent the coefficients of the quadratic equation $ax^2 + bx + c = 0$. Read the values for **a**, **b**, and **c** from the user. Then use if- else if - else statements to output how many solutions the quadratic equation has and also print the solutions; if there is any.
4. An employee is paid at a rate of \$16.78 per hour for the first 40 hours worked in a week. Any hours over that are paid at the overtime rate of one and-one-half times that. From the worker's gross pay, 6% is withheld for Social Security tax, 14% is withheld for federal income tax, 5% is withheld for state income tax, and \$10 per week is withheld for union dues. If the worker has three or more dependents, then an additional \$35 is withheld to cover the extra cost of health insurance beyond what the employer pays. Write a program that will read in the number of hours worked in a week and the number of dependents as input and will then output the worker's gross pay, each withholding amount, and the net take-home pay for the week.
5. Write a program that declares five integer variables, reads in five integers and then outputs the sum of only the positive inputs, the average of only the positive inputs, the sum of only the negative inputs, the average of only the negative inputs, the sum of all the inputs, and the average of all the inputs.
6. Write a C++ program that calculates and prints a random integer in the range [-1, 1].
7. Write a C++ program that calculates and prints a random float in the range [-1.0, 1.0)
8. Write a C++ program that reads two integers **x** and **y**. Then swaps their values if necessary so that to make sure **y** \geq **x**. Then your program must calculate and print a random integer in the range [**x**, **y**].

9. Write a C++ program that reads two floats **x** and **y**. Then swaps their values if necessary so that to make sure **y** \geq **x**. Then your program must calculate and print a random float in the range [**x**, **y**).
10. Write a C++ program that reads one character from the user and then prints the message "The character you entered is a digit" if the character is one of the digits '0', '1', '2', ... '9'; prints the message "The character you entered is a lower case" if the character is one of the lower case English alphabets 'a', 'b', 'c', ... 'z'; prints the message "The character you entered is an upper case" if the character is one of the upper case English alphabets 'A', 'B', 'C', ... 'Z'; otherwise the program must print "The character you entered is neither an English alphabet nor a digit."
11. **[Challenge]** Write a C++ program that prints **randomly** one of the following three messages: **Yes**, **No** or **Not-Sure**.
12. **Which Pizza to buy!** The large "economy" size of an item is not always a better buy than the smaller size. This is particularly true when buying pizzas. Pizza sizes are given as the diameter of the pizza in inches. However, the quantity of pizza is determined by the area of the pizza, and the area is not proportional to the diameter. Most people cannot easily estimate the difference in area between a 10-inch pizza and a 12-inch pizza and so cannot easily determine which size is the best buy—that is, which size has the lowest price per square inch.

Write a C++ program that does the following:

- Declares the four float variables size_smaller, size_larger, price_smaller, and price_larger,
- Reads the size and price of a smaller pizza in the variables size_smaller and price_smaller,
- Reads the size and price of a larger pizza in the variables size_larger and price_larger,
- Declare two float variables area_smaller and area_larger
- Compute the area of each pizza in the variables area_smaller and area_larger. Use the value 3.14 for pi
- Compute the price per square inch for each pizza and determine which pizza is a better buy.

NOTE: The pizza that has lower price per square inch is the better buy.

13. Write a C++ program that reads in three double data type values from the user and then prints the maximum and the minimum among the three user input values.

14. Simple Interest Calculator

Write a C++ program that calculates the accrued total amount given a principal amount and time period in years. The accrued total amount is calculated as

$$\text{totalAmount} = \text{principalAmount} * (1.0 + \text{rateOfInterest} * \text{timePeriod})$$

For the rate of interest use the following table:

Principal Amount	Rate of Interest
[0.00, 1000.00)	2.5%
[1000.0, 10,000.00)	2.0%
[10,000.00, 100,000.00)	1.5%
>= 100,000.00	1.0%

The program will proceed as follows:

- Ask the user for the principal amount
- Ask the user for the number of years
- If principal amount is less than 0 or the number of years is less than 0, then print an error message and do nothing
- Else, calculate the accrued total amount and print it

15. Write a C++ program that generates a random upper case English alphabet character and that prints the upper case English alphabet character that is ten letters away from the character generated. For example,

If the random character is 'A' then your program must print 'K'
If the random character is 'B' then your program must print 'L'
If the random character is 'C' then your program must print 'M'
If the random character is 'Q' then your program must print 'A'
If the random character is 'V' then your program must print 'F'
If the random character is 'Z' then your program must print 'J'

Observe that if the character that is ten letters away from the random character is outside of the upper case English letters; then we must go back to 'A'.

16. Area of Triangle Given the three sides of a triangle, the general formula to calculate the area of the triangle is

$$\text{area} = \sqrt{s(s-a)(s-b)(s-c)}$$

where a, b, and c are the sides of the triangle and $s = \frac{a+b+c}{2}$

Write a C++ program that reads three float numbers (a, b, and c) and prints the area of the triangle formed by the sides a, b, and c. You must first check if triangle inequality holds;

that is the sum of any two sides is greater than the third. If triangle inequality doesn't hold; then your program should print error message and do nothing.

17. Vector Length Given two points on a plane $P1(x1, y1)$ and $P2(x2, y2)$; the length of the line segment connecting $P1$ and $P2$ is given by $d = \sqrt{(x1 - x2)^2 + (y1 - y2)^2}$. Write a C++ program that reads four double numbers $x1, y1, x2$, and $y2$ and prints the length of the line segment connecting the two points $P1(x1, y1)$ and $P2(x2, y2)$.

18. Area of Triangle Consider the area of a triangle whose three vertices on the plane are given as points $P1(x1,y1)$, $P2(x2,y2)$, and $P3(x3,y3)$. Write a C++ program that reads six double type numbers $x1, y1, x2, y2, x3$, and $y3$ and prints the area of the triangle. You must first check if triangle inequality holds; that is the sum of any two sides is greater than the third. If triangle inequality doesn't hold; then your program should print error message and do nothing.

19. Do You Want To Play Blackjack? Write a program that scores a **SIMPLE** blackjack hand. In **SIMPLE** blackjack, a player receives **3** cards where each card is one of the digits 2 through 9, 10, Jack, Queen, King, or Ace. The cards 2 through 10 are scored as 2 through 10 points each. The face cards - Jack, Queen, and King - are scored as 10 points. The goal is to come as close to a score of 21 points as possible without going over 21 points. The Ace card can be counted as either 1 or 11 points whichever is better for the player.

For example an Ace, J, and 10 cards can be scored as either 31 or 21 points. Since 21 is a better score, this hand is scored as 21. On the other hand an Ace, 5 and 2 cards can be scored as either 18 or 8 points. Since 18 is a better score, this hand is scored as 18.

Assume the **3** cards at hand are represented as character data types. The cards 2 through 9 are represented as the characters '2' through '9'; while the cards 10, Jack, Queen, King, and Ace are represented respectively as the characters 'T', 'J', 'Q', 'K', and 'A'.

Your program should first read in the values of three character variables representing the **3** cards from the user and then convert the card values from character values to numeric score points. Your program should then compute the best sum taking care of Aces. Finally your program must print:

The message "Busted" if the sum is over 21.

The message "Black Jack" if the sum is exactly equal to 21.

The sum value if it is less than 21.

HINT:- First consider the Aces as if they score 1 point each and compute the sum.

Afterwards adjust the sum taking care of the Aces.

20. Repeat Q#19 above so that this time you **do not read in the three cards from the user**; but rather **pick three random cards** from '2', '3', '4', ..., '9', 'T', 'J', 'Q', 'K' or 'A'.