CE 100 - CS 101 - CS 112 - CS 203 Eng. Nageh Mahmoud 65651669

CS 101 MIDTERM SUMMARY

CHAPTER 4: MAKING DECISIONS

CHAPTER 4: Making Decisions

4.1 Relational Operators

Relational operators allow you to compare numbers, characters, and strings to determine equality or order. They evaluate to a Boolean value (`true` / `1` or `false` / `0`).

Operator	Meaning	Example
`>`	Greater than	`x > y`
`<`	Less than	`x < y`
`>=`	Greater than or equal to	`x >= y`
`<=`	Less than or equal to	`x <= y`
`==`	Equal to	`x == y`
`!=`	Not equal to	`x != y`

REMEMBER: The == operator determines relational equality, whereas the = operator assigns a value to a variable.

Question: Given x = 10 and y = 7, what is the evaluated value for each of the following relational expressions?

```
int z = (x < y);  // z = 0 (false)
cout << (x > y);   // Displays 1 (true)
int a = (x >= y);  // a = 1 (true)
cout << (x <= y);  // Displays 0 (false)
int b = (y != x);  // b = 1 (true)
```

4.2 Comparing Chars and Strings

- **Comparing Char:** Characters are compared using ASCII integer values ('0'-'9': 48-57, 'A'-'Z': 65-90, 'a'-'z': 97-122).
- **Comparing Strings:** Compared character-by-character from left to right using their underlying ASCII values.

Question: Given str1 = "Hello", str2 = "Hi", str3 = "Air", and str4 = "Bill", determine whether each string comparison evaluates to true or false.

```
str1 < str2  // true ('e' < 'i')
str1 > "Hen" // false ('l' < 'n')
str3 < "An"  // true ('i' < 'n')
str1 == "hello" // false ('H'=72 != 'h'=104)
str3 <= str4 // true ('A' < 'B')
str2 > str4  // true ('H' > 'B')
```

4.3 Logical Operators

Operator	Name	Explanation
&&	AND	Evaluates to `true` only if both conditions are true.
||	OR	Evaluates to `true` if at least one condition is true.
!	NOT	Reverses truth (`!true` = `false`, `!false` = `true`).

Question: Write C++ code snippets demonstrating how to use AND (&&), OR (||), and NOT (!) logical operators in temperature checks.

```
if (temperature < 20 && minutes > 12)
    cout << "The temperature is in the danger zone.";

if (temperature < 20 || temperature > 100)
    cout << "The temperature is in the danger zone.";

if (!(temperature > 100))
    cout << "You are below the maximum temperature.
";
```

4.4 Decision Structures

A. The `if` Statement

Question: Write a complete C++ program to prompt the user for the length and width of a rectangle and calculate its area.

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

int main() {
    double length, width, area;
    cout << "Enter length: ";
    cin >> length;
    cout << "Enter width: ";
    cin >> width;
    area = length * width;
    cout << "The area is: " << area << endl;
    return 0;
}
```

B. The `if/else` Statement

Question: Write an if/else statement to check whether an integer is even or odd and print the result.

```
if (number % 2 == 0)
    cout << "The number is even.";
else
    cout << "The number is odd.";
```

C. Nested `if` Statements

Question: Write nested if statements to check if an applicant is employed ('Y') and a recent graduate ('Y') to qualify for a special loan rate.

```
if (employed == 'Y') {
    if (recentGrad == 'Y') {
        cout << "You qualify for the special interest rate.
";
    } else {
        cout << "You must have graduated from college in the past two years to qualify.
";
    }
} else {
    cout << "You must be employed to qualify.
";
}
```

D. Multi-Branch `if / else if / else` Ladder

Question: Write an if/else if ladder to determine a student's letter grade (A, B, C, D, or F) based on their final score.

```
if (finalScore >= 90)
    cout << "You got A!" << endl;
else if (finalScore >= 80)
    cout << "You got B!" << endl;
else if (finalScore >= 70)
    cout << "You got C!" << endl;
else if (finalScore >= 60)
    cout << "You got D!" << endl;
else
    cout << "You got F!" << endl;
```

E. Weather App & Ticket Pricing Examples

Question: Write an else-if structure for a Weather App to output clothing recommendations, and for a Ticket App based on user age.

```
// Weather App
double temperature = 15.5;
if (temperature < 0) {
    cout << "It's freezing!";
} else if (temperature < 20) {
    cout << "Might need a sweater.";
} else {
    cout << "It's warm outside!";
}

// Ticket App
int age = 17;
if (age < 13) {
    cout << "Child ticket";
} else if (age < 20) {
    cout << "Teen ticket";
} else {
    cout << "Adult ticket";
}
```

4.5 Flags

Question: How do you use a boolean variable as a flag to track if sales quota was met, and conditionally print a message?

```
bool salesQuotaMet;
if (sales >= QUOTA_AMOUNT)
    salesQuotaMet = true;
else
    salesQuotaMet = false;

if (salesQuotaMet)
    cout << "You have met your sales quota!
";
```

4.6 The `switch` Statement

Question: Write a switch statement to display performance feedback based on a letter grade ('A' through 'F').

```
char grade;
cin >> grade;
switch (grade) {
    case 'A': cout << "Great job!!"; break;
    case 'B': cout << "Good job"; break;
    case 'C': cout << "Satisfactory job"; break;
    case 'D': cout << "Need to work a little harder"; break;
    case 'F': cout << "Sorry, you failed the class"; break;
    default:  cout << "Invalid grade";
}
```

Question: Write a simple calculator program using a switch statement to perform +, -, *, and / operations on two numbers.

```
double num1, num2, result;
char operation;
cout << "Enter operator (+, -, *, /): ";
cin >> operation;
cout << "Enter two numbers: ";
cin >> num1 >> num2;

switch (operation) {
    case '+': result = num1 + num2; break;
    case '-': result = num1 - num2; break;
    case '*': result = num1 * num2; break;
    case '/':
        if (num2 != 0) result = num1 / num2;
        else cout << "Error: Division by zero!";
        break;
    default: cout << "Invalid operator.";
}
```

Question: What is the output of the following switch statement when break statements are omitted (switch fall-through)?

```
int day = 1;
switch (day) {
    case 1: cout << "Monday.
";
    case 2: cout << "Tuesday.
";
    case 3: cout << "Wednesday.";
}
// Output:
// Monday.
// Tuesday.
// Wednesday.
```

4.7 Conditional / Ternary Operator

Question: Rewrite the statement "if (x < 0) y = 10; else z = 20;" using the ternary conditional operator.

```
x < 0 ? y = 10 : z = 20;
```