

Lab 5 – if-else and switch Statements

Topics to be covered:

- if -else statements
- switch statements

Example code:

- IfElseSwitch.cpp

Code Segments (Entire code is in .cpp file above):

```
***** IfElseSwitch.cpp *****
//if-else statements
//Curly braces are optional if it is single statement.
if (intChoice == 1) cout << "You chose: Cheesecake." << endl;
else if (intChoice == 2 || intChoice == 6)
    cout << "You chose: Chocolate Pie." << endl;
else if (intChoice >= 3 && intChoice <= 5)
    cout << "You chose: Icecream." << endl;

//A pair curly braces is needed for multiple statements.
else
{
    cout << "OOPS... no matching" << endl;
    cout << "Welcome to come again." << endl;
}

//switch statements
switch (charChoice)
{
    case 'a':
    case '1':
        cout << ... ;
        break;

    case 'b':
    case '2':
        ...
        break;

    default:
        ...
}
```

Lab 5

Exercise 1

Write a defining diagram and pseudocode for Assignment 1. *You'll need to use IF, IF-ELSE and CASE pseudocode formats.*

Assignment 1

Write a complete C++ program that presents the user with a menu of 3 choices on the screen. The user should be able to select any one of the 3 selections and the program should call an appropriate function to process the selection. **Use a switch statement** to process the menu selection.

Selections are:

1. Some arithmetic
2. Fun with letters
3. Play with words

If an invalid selection is made respond with an error message

Each of the selections should work as follows:

1. prompt the user for 2 integers and an arithmetic operator
 - call a function sending it the 2 ints and one char as parameters
 - the function should calculate and display the result of the arithmetic operation on the 2 ints. The only valid operators will be +, -, *, /.
 - i. Make sure to check for and handle division by 0.
2. prompt the user for a letter
 - check to make sure the entry is a valid letter: 'a' to 'z' or 'A' to 'Z'
 - i. display an error message if not
 - if valid and uppercase (lowercase)
 - i. call a function that returns lowercase (uppercase) equivalent
 - display the two letters (upper and lower case) and the ASCII value of both letters to the screen.
3. Call a function that opens an input file "name.txt"
 - reads *last name* followed by *first name* and writes to the screen the statement:
"Hello there ***first name last name***, have a nice day."
 - i. each name will not exceed 25 characters
 - ii. note the order read/written