University of Makati
College of Computing and Information Sciences

OPERSYS
Week 4 Activity

Members:

Armojallas, Caleb Joshua
Canlas, Korbin
Manuel, John Lou
Santos, Meinard Edrei

1. Creating and Executing a Batch File

A batch file is a simple script that contains a series of commands executed in sequence by the command-line interpreter. To create a batch file, you can follow these steps:

- **Create a New File:** Open any text editor like Notepad.

Write the Script: Type your commands in the editor. For example:

```
@echo off
echo Hello, World!
pause
```

- In this example, `@echo off` hides the command that appears in the console, `echo` displays a message, and `pause` waits for a user input before closing the window.
- **Save the File:** Save the file with a `.bat` extension, such as `script.bat`.
- **Execute the File:** Double-click the `.bat` file, or run it via the Command Prompt by navigating to its directory and typing `script.bat`.

This simple structure allows batch files to automate repetitive tasks or command sequences, making them highly useful for system administration.

2. Command Line Arguments in Batch Files

Batch scripts can accept command line arguments to pass information when executing. These arguments are accessed using %1, %2, %3, etc., depending on their position.

For example:

```
@echo off

echo First argument: %1

echo Second argument: %2

pause
```

When running `script.bat Hello World`, the output will be:

```
First argument: Hello

Second argument: World
```

This ability to pass arguments makes batch files flexible for various tasks, like file manipulation or running programs with dynamic inputs.

3. Defining Variables to Hold Numeric Values

In batch scripting, you can define variables to hold numeric values using the `set` command. Batch scripts inherently treat all variables as strings, but you can perform basic arithmetic operations using the `set /a` command.

Example:

```
@echo off

set /a number=5

set /a result=number*2

echo Result is: %result%

pause
```

In this example, the `/a` flag allows the script to treat the variables as numbers and perform arithmetic, which is useful for loops, counters, and other numeric-based operations.

4. Working with Strings in Batch Script

Strings in batch scripts are handled as text data. You can define a string variable, concatenate them, or extract substrings. String variables are created using the `set` command.

Example:

```
@echo off

set name=John

echo Hello, %name%
```

You can also perform string manipulations, such as removing substrings or getting the length of a string, though batch script has limited built-in string functions compared to more advanced scripting languages.

For instance, you can extract a substring using:

```
set myString=HelloWorld

echo %myString:~0,5%
```

This prints "Hello", extracting the first five characters from `myString`.

5. Other Features of Batch File Scripting

Batch scripting offers many useful features for automating tasks:

Conditional Statements: You can use `if` statements for conditional execution.

```
if exist myfile.txt echo File exists
```

- This checks if a file exists and executes a command based on the condition.

Loops: The `for` loop allows iteration over files, directories, or lists.

```
for %%x in (*.txt) do echo %%x
```

- This loops through all `.txt` files in the current directory and echoes their names.

Error Handling: Use `errorlevel` to capture and react to errors.

```
if %errorlevel% neq 0 echo Error occurred
```

Comments: You can add comments using `rem` to make your script more readable.

```
rem This is a comment
```

Conclusion

Batch scripting is a powerful tool for automating tasks in Windows environments. It supports command line arguments, string and numeric manipulation, loops, and conditional statements. The ease of creating and executing batch files makes them an essential part of system automation and task scheduling. With its flexibility, batch scripting can significantly enhance productivity, especially in system administration and file management tasks.