Introduction to CLI Interfaces
(and a bit about me...)

Deepak Kumar

A Mainframe Computer

IBM 1401
A “USB” Cable!
The IBM 1403 Printer at Work

A Microcomputer

What’s missing???
Computer Interfaces

• **Graphics User Interface (GUI)**
  aka **WIMP Interface (Windows, Icons, Mouse, Pointing)**

Users move a mouse-pointer over icons and use mouse-clicks to carry out computing tasks.

• **Command Line Interface (CLI)**

User types in text commands to carry out computing tasks: navigating files, folders, running programs and services, etc.

A CLI is called a **shell** (why?)
Re-learning to use a computer

- Using a CLI shell – **Git Bash**

---

Basic Commands we will learn

- `cat`
- `cd`
- `cp`
- `less`
- `ls`
- `mkdir`
- `more`
- `pwd`
- `rm`
- `whoami`
whoami

• This command outputs the name of the user that is currently logged in (you!)

  ~> whoami
dkumar

pwd

• Present Working Directory

  This command responds with the complete path of the current directory.

  ~> pwd
  /c/users/dkumar

  or
  ~> pwd
**Mkdir <name>**

- Make a new directory called <name> in the current one

```
~> mkdir Test
~>
```

**ls**

- List the contents of the present directory

```
~> ls
  abc.txt  hello.java
  letters  mail/  Test/
```
Cd <location/path>

• Change directory to <location/path> specified

~~> cd Test
~/Test>

~/Test> cd ..  [.. Stands for Parent directory]
~~>

~~> cd ~  [~ Stands for Home directory]
~~>

cat/more/less <filename>

• Show contents of <filename> specified

~~> cat Hello.java
class HelloWorld {
   public static void main(String[] args) {
      System.out.println("Hello, World!");
   } // main()
} // class HelloWorld
~~>
cp <file1> <file2>

- Make a copy of <file1> to <file2>

```bash
~> cp Hello.java Aloha.java
~> ls
Hello.java  Aloha.java
~>
```

rm <filename>

- Remove the <filename> specified

```bash
~> rm Hello.java
~>
```
javac-introcs <filename.java>

- Compile the Java file <filename.java>
- If successful, creates a file <filename.class>
- Otherwise, errors in Java file are reported

```bash
~> javac-introcs Hello.java
~> ls
Hello.class  Hello.java
~>
```

java-introcs <file>

- Run the file <file>.class

Note: The name of the file should match the name of class in Java file

```bash
~> java-introcs Hello
Hello, World!
~>
```
Picture Credits

- Pictures on slides 2 through 6 are taken by and property of Deepak Kumar.
- Pictures on slides 2-5 were taken at the Computing History Museum, Mountain View, CA in Feb 2019.
- Picture of VT 100 Terminal on Slide 7 is by Jason Scott. Downloaded from Wikipedia (https://commons.wikimedia.org/wiki/File:DEC_VT100_terminal.jpg) on September 4, 2020. Licensed under Creative Commons.