2.1 Interacting with Computers

- Interactive: Accepting input from a human user
  - Interactive computer system: Programs that allow users to enter data or commands
  - Non-interactive system: Programs that when started, continues without requiring human contact. e.g. compiler
- Goals of interaction between user and system:
  - Effective operation and control of the machine (successful in producing desired or intended result)
  - Feedback from the system, aiding user in making operation decisions

- **Input-Output (I/O) Devices**
  - Provides a medium for interactivity
  - Input devices
    - Provides data and control signals to computer
    - Capture user inputs e.g. texts, sounds, movements, graphics, gestures
    - Devices: keyboard, mouse, microphone, scanner, camera
  - Output devices
    - Converts machine-readable information to human-readable form
    - Devices: monitor/display, speaker, printer

- **User Interfaces**
  - Input devices and software
  - Usability: ease of use and learnability
  - Poor User Interfaces:
    - Features:
      - Lack of order
      - Confusing
      - Takes time to understand
      - Trail and error
    - Consequences:
      - Require more time to perform tasks
      - Higher chances of making errors
      - Feels dissatisfied
      - Require more time to learn to use program

        - May refrain from using program
    - Good User Interfaces:
        - Features:
            - Attractive
            - Intuitive:
                - Point & click
                - Icon takes on meaning of action/command
            - Consequences:
                - Ease of use & learnability
                - Allows for higher productivity

- **Types of Interfaces**
    - **Graphical User Interface (GUI):**
        - Visual way of interacting with computer using items e.g. windows, icons, menus
        - Click/select with pointing device instead of typing command
        - Direct manipulation of graphics elements
        - WMIP (window, menu, icon, paradigm)
    - **Command Line Interface (CLI):**
        - Accept only predefined command names
        - Not all things are predefined
        - No help given unless command input by user
        - E.g. cmd in windows, terminal in macOS, DOS
        - Strengths:
            - Works faster if user knows the command
            - Complete control over system
            - Ability to perform complex operations
            - Powerful for many tasks e.g. deal with many files
            - Less memory used
            - Allows users to perform one or more commands using one simple, easy to write command
    - **Menu Based Interface (MBI):**
        - Offers list of options which user can choose from
        - Allows user to select correct command from list
        - Commonly found on applications, monitors, TVs
        - User interacts by choosing correct/preferred options
        - No help necessary
        - Limited and predefined functions
        - Strengths:
            - Easy to use

        - Nothing to remember
        - User can't perform pre-made functions, won't destroy the system
    - Limitations:
        - Confined by input methods
        - Limit input choices
- **Other UIs:**
    - Touchscreen:
        - Similar to GUI, but without the input devices
        - Used in: ATM, smartphones
- **Gesture Interface:**
    - Gestures —> smaller scale and more specific form of motion
    - Involves interpretation of human gestures and using them as input data for computer system
    - Focuses mainly on the arm, hand and facial gestures
    - Require devices to capture ranging from wired gloves to camera
    - e.g. mobile phone OS
    - Strengths:
        - Intuitive, ease of use, due to gestures resemble real life actions
        - Touch screen gives user better control
    - Challenges:
        - Accuracy of tracking
        - Requires conducive environment without background distractions
    - Areas of use:
        - Aiding disabilities: sign language recognition; control through facial gestures
        - Productivity: virtual controllers
        - Leisure: gaming
- **Motion Tracking Interface:**
    - Monitor the user's body motions and translate them into commands
- **Voice User Interface:**
    - Accept inputs and provides outputs by generating voice prompts
    - User input is made by pressing keys/buttons, or responding verbally to the interface
    - Strengths:

        - Shifts focus from visual component to audio component
        - Better productivity as it’s hands-free
    - Challenges:
        - Background noise, requires conducive environment
        - Ambiguous inputs
    - Areas of use:
        - Personal assistant e.g. Siri, Cortana
- **Natural Language Interfaces:**
    - Users can use without conscious attention to the interface
    - Input and output of device in our "everyday" language
    - System interprets command given by user
    - Sought after their speed and ease of use
    - Suffer the challenges to understand wide varieties of ambiguous inputs
    - E.g. Wolfram, Google
    - Strengths:
        - Does not require any knowledge as user would know basic communication
    - Challenges:
        - Complex algorithms to decode input
        - Ambiguity
        - Natural isn’t defined, differs by people

- **Design Considerations for User Interfaces**
    - **Format:**
        - Layout
        - Colour and texture
        - Imagery
        - Animation
        - Sequencing
        - Sound
        - Visual identity
    - **Eight Golden Rules of Design:**
        1. Consistency
            - Developing usage pattern
            - Consistent sequence and designs
            - Identical terminology
        2. Use of shortcuts
            - Reduce number of interactions, increase efficiency

        - Function keys
        - Hidden commands
        - Macros
3. Informative feedback
    - Allows users to know what's happening
4. Dialog for closure
    - Gives user sense of accomplishment
    - Indicate ready for next action
    - Sequence of actions organised into groups with start, middle, end
5. Simple error handling
    - Design system so user can't make serious error
    - System to detect errors made and offer simple mechanisms for handling error
6. Easy reversal of actions
    - Encourages exploration of unfamiliar actions as users know errors can be undone
        - Undo
        - Recovery
7. Internal locus of control
    - Gives user sense that they are in charge and system responds to their actions
8. Reduce short-term memory load
    - Humans have limitations in information processing in short-term memory
    - Information must be clear and concise
- *There will be conflicts in implementing all the rules, must find balance depending on the system/program