Class Note: Flowcharts, Pseudocodes, and Selection Constructs

Session 3: Flowcharts and Pseudocodes

Overview

This session introduces the foundational tools for algorithm design: **flowcharts** and **pseudocodes**. These tools help programmers plan and visualize solutions to problems before writing actual code, ensuring clarity and logical structure.

Key Concepts

1. **Algorithm**:
   - A step-by-step procedure to solve a problem.
   - Characteristics: Finite steps, clear instructions, produces a result.
   - Example: A recipe for making a cake is an algorithm with steps like "mix flour and sugar" and "bake at 350°F."
2. **Flowcharts**:
   - A visual representation of an algorithm using standard symbols:
     - **Oval**: Start/End of the process.
     - **Rectangle**: Process or action step.
     - **Diamond**: Decision point (e.g., yes/no question).
     - **Arrow**: Flow of control.
     - **Parallelogram**: Input/Output operation.
   - Flowcharts help visualize the sequence of steps and decision-making in an algorithm.
   - Example: A flowchart to decide if a number is even or odd involves a decision diamond to check if the number is divisible by 2.
3. **Pseudocode**:
   - A high-level, human-readable description of an algorithm using plain language and structured syntax.
   - Not tied to any specific programming language, making it universal for planning.
   - Common keywords: `START`, `END`, `INPUT`, `OUTPUT`, `IF`, `WHILE`, `FOR`.
   - Example: Pseudocode to calculate the average of two numbers:

```
START
INPUT number1, number2
sum = number1 + number2
average = sum / 2
OUTPUT average
END
```

4. **Benefits**:

- Flowcharts and pseudocodes simplify complex problems, aid debugging, and improve communication among programmers.
- They serve as a blueprint before coding, reducing errors in implementation.

Examples

1. **Flowchart Example: Check if a Number is Positive**:

   - **Description**: Create a flowchart to determine if a user-entered number is positive, negative, or zero.
   - **Flowchart**:
     - Start (Oval)
     - Input number (Parallelogram)
     - Decision: Is number > 0? (Diamond)
       - Yes: Output "Positive" (Parallelogram) → End (Oval)
       - No: Decision: Is number = 0? (Diamond)
         - Yes: Output "Zero" (Parallelogram) → End (Oval)
         - No: Output "Negative" (Parallelogram) → End (Oval)
   - **Visual Representation** (described for clarity):

```
[Start] → [Input number] → [number > 0?] → Yes → [Output "Positive"] → [End]
                                ↓ No
                       [number = 0?] → Yes → [Output "Zero"] → [End]
                                ↓ No
                       [Output "Negative"] → [End]
```

2. **Pseudocode Example: Calculate Area of a Rectangle**:

   - **Description**: Write pseudocode to calculate the area of a rectangle given its length and width.
   - **Pseudocode**:

```
START
INPUT length, width
area = length * width
OUTPUT "The area is ", area
END
```

3. **Combined Example: Grading System**:
   - **Description**: Design an algorithm to assign a letter grade based on a test score (0–100).
   - **Pseudocode**:

```
START
INPUT score
IF score >= 90 THEN
    OUTPUT "Grade A"
ELSE IF score >= 80 THEN
    OUTPUT "Grade B"
ELSE IF score >= 70 THEN
    OUTPUT "Grade C"
ELSE IF score >= 60 THEN
    OUTPUT "Grade D"
ELSE
    OUTPUT "Grade F"
END IF
END
```

   - **Flowchart** (described):
     - Start → Input score → Decision: score ≥ 90? → Yes: Output "Grade A" → End
     - No → Decision: score ≥ 80? → Yes: Output "Grade B" → End
     - No → Decision: score ≥ 70? → Yes: Output "Grade C" → End
     - No → Decision: score ≥ 60? → Yes: Output "Grade D" → End
     - No → Output "Grade F" → End

Classwork Activity 1: Flowcharts and Pseudocodes

1. **Flowchart Design**:
   - Draw a flowchart to determine if a person is eligible to vote (age ≥ 18).
   - Include symbols for Start, Input (age), Decision (age ≥ 18), Output ("Eligible" or "Not Eligible"), and End.
2. **Pseudocode Writing**:
   - Write pseudocode to calculate the total cost of items in a shopping cart:
     - Input: Price and quantity of an item.

- ○ Calculate: Total = price * quantity.
- ○ Output: Total cost.

3. **Group Activity**:

- In pairs, create a flowchart and pseudocode for a program that checks if a year is a leap year (divisible by 4, but not by 100 unless also divisible by 400).
- Present your solution to the class and discuss any differences in approaches.

Session 4: Selection Constructs

Overview

Selection constructs allow programs to make decisions based on conditions, enabling different outcomes depending on input. This session covers **if-then**, **if-then-else**, and **nested if** statements, which are essential for controlling program flow.

Key Concepts

1. **Selection Constructs**:

- Programs use conditions to choose between different paths of execution.
- Common constructs:
    - ○ **If-Then**: Execute a block of code if a condition is true.
    - ○ **If-Then-Else**: Execute one block if true, another if false.
    - ○ **Nested If**: Conditions within conditions for complex decision-making.
- Conditions often use relational operators (e.g., >, <, ==, ≠) and logical operators (e.g., AND, OR, NOT).

2. **If-Then Statement**:

- Syntax (in pseudocode):

```
IF condition THEN
    action
END IF
```

- Example: If a temperature is above 30°C, display "It's hot."

3. **If-Then-Else Statement**:

- Syntax:

```
IF condition THEN
    action1
ELSE
    action2
END IF
```

- Example: If a number is even, display "Even"; otherwise, display "Odd."

4. **Nested If Statement**:
   - Syntax:

```
IF condition1 THEN
    IF condition2 THEN
        action1
    ELSE
        action2
    END IF
ELSE
    action3
END IF
```

   - Example: Check if a student passes (score ≥ 60) and qualifies for honors (score ≥ 85).

5. **Logical and Relational Operators**:
   - **Relational**: >, <, ≥, ≤, ==, ≠
   - **Logical**: AND (both true), OR (at least one true), NOT (negates condition)
   - Example: `IF age ≥ 18 AND citizen == "Yes" THEN` checks voting eligibility.

Examples

1. **If-Then Example: Check Temperature**:
   - **Pseudocode**:

```
START
INPUT temperature
IF temperature > 30 THEN
    OUTPUT "It's hot outside"
END IF
END
```

   - **Flowchart** (described):
     - Start → Input temperature → Decision: temperature > 30? → Yes: Output "It's hot outside" → End

○ No → End

2. **If-Then-Else Example: Even or Odd Number**:
    - **Pseudocode**:

    ```
    START
    INPUT number
    IF number MOD 2 == 0 THEN
        OUTPUT "Number is Even"
    ELSE
        OUTPUT "Number is Odd"
    END IF
    END
    ```

    - **Flowchart** (described):
        ○ Start → Input number → Decision: number % 2 == 0? → Yes: Output "Even" → End
        ○ No → Output "Odd" → End

3. **Nested If Example: Student Grading with Honors**:
    - **Pseudocode**:

    ```
    START
    INPUT score
    IF score >= 60 THEN
        IF score >= 85 THEN
            OUTPUT "Pass with Honors"
        ELSE
            OUTPUT "Pass"
        END IF
    ELSE
        OUTPUT "Fail"
    END IF
    END
    ```

    - **Flowchart** (described):
        ○ Start → Input score → Decision: score ≥ 60? → Yes: Decision: score ≥ 85? → Yes: Output "Pass with Honors" → End
        ○ No (inner): Output "Pass" → End
        ○ No (outer): Output "Fail" → End

4. **Complex Example with Logical Operators: Driving Eligibility**:
    - **Description**: Check if a person can drive based on age (≥ 16) and having a license.
    - **Pseudocode**:

```
START
INPUT age, hasLicense
IF age >= 16 AND hasLicense == "Yes" THEN
    OUTPUT "Eligible to drive"
ELSE
    OUTPUT "Not eligible to drive"
END IF
END
```

- **Flowchart** (described):
  - Start → Input age, hasLicense → Decision: age ≥ 16 AND hasLicense == "Yes"? → Yes: Output "Eligible to drive" → End
  - No → Output "Not eligible to drive" → End

Classwork Activity 2: Selection Constructs

1. **Pseudocode Writing**:
   - Write pseudocode to check if a user's age qualifies them for a discount:
     - Senior discount: age ≥ 65
     - Student discount: age 13–22
     - No discount: otherwise
2. **Flowchart Design**:
   - Create a flowchart for a program that determines if a customer gets free shipping:
     - Free shipping if order amount > $50 or customer is a premium member.
     - Otherwise, charge $5 shipping.
3. **Error Correction**:
   - Given the pseudocode:

   ```
   INPUT score
   IF score > 60 THEN
       OUTPUT "Pass"
   ELSE score <= 60 THEN
       OUTPUT "Fail"
   END IF
   ```

   Identify and fix the error (redundant condition in ELSE).
4. **Group Activity**:
   - In pairs, design a flowchart and pseudocode for a program that determines if a person can enter a movie theater based on:
     - Age ≥ 18 OR accompanied by an adult (hasAdult = "Yes").

 o Present your solution and discuss alternative approaches.

Objective Questions (Multiple Choice)

Test your understanding of Sessions 3 and 4 with the following questions:

1. What is the purpose of a flowchart in programming?
   - A) To execute code directly
   - B) To visualize an algorithm's steps
   - C) To compile code into machine language
   - D) To store data permanently
   - **Answer**: B
2. Which symbol in a flowchart represents a decision point?
   - A) Rectangle
   - B) Oval
   - C) Diamond
   - D) Parallelogram
   - **Answer**: C
3. In pseudocode, what keyword is used to mark the beginning of an algorithm?
   - A) BEGIN
   - B) START
   - C) INIT
   - D) OPEN
   - **Answer**: B
4. Which of the following is a valid pseudocode for calculating the square of a number?
   - A)

     ```
     INPUT num
     square = num * num
     OUTPUT square
     ```

   - B)

     ```
     INPUT num
     OUTPUT num * num
     square = num
     ```

   - C)

```
    square = INPUT num
    OUTPUT num
```

- D)

```
    num = square * square
    OUTPUT num
```

- **Answer**: A

5. What does the following pseudocode do?

```
INPUT number
IF number > 0 THEN
    OUTPUT "Positive"
END IF
```

- A) Checks if a number is negative
- B) Outputs "Positive" if the number is greater than 0
- C) Calculates the square of the number
- D) Outputs "Positive" for all numbers
- **Answer**: B

6. In a selection construct, what happens if the condition in an IF-THEN statement is false?
- A) The program terminates
- B) The THEN block is executed
- C) The program skips the THEN block
- D) An error occurs
- **Answer**: C

7. Which operator is used to combine two conditions so both must be true?
- A) OR
- B) AND
- C) NOT
- D) XOR
- **Answer**: B

8. What is the output of the following pseudocode for input score = 75?

```
IF score >= 70 THEN
    OUTPUT "Pass"
ELSE
    OUTPUT "Fail"
END IF
```

- A) Pass
- B) Fail
- C) No output
- D) Error
- **Answer**: A

9. In a nested IF statement, how many conditions are evaluated if the outer condition is false?

- A) All inner conditions
- B) None
- C) Only the first inner condition
- D) Only the ELSE block
- **Answer**: B

10. Which flowchart path correctly represents the pseudocode:

```
IF age >= 18 THEN
    OUTPUT "Adult"
ELSE
    OUTPUT "Minor"
END IF
```

- A) Input age → Decision: age ≥ 18? → Yes: Output "Adult" → End; No: Output "Minor" → End
- B) Input age → Output "Adult" → Decision: age ≥ 18? → End
- C) Input age → Decision: age < 18? → Yes: Output "Adult" → End
- D) Input age → Output "Minor" → End
- **Answer**: A

Homework

1. **Flowchart and Pseudocode**:
   - Create a flowchart and pseudocode for a program that determines if a number is divisible by both 3 and 5.
2. **Selection Construct Practice**:
   - Write pseudocode to categorize a person's BMI (Body Mass Index):
     - Input: weight (kg), height (m).
     - Calculate: BMI = weight / (height * height).
     - Categories: Underweight (< 18.5), Normal (18.5–24.9), Overweight (25–29.9), Obese (≥ 30).
3. **Real-World Application**:

- Research a real-world scenario where selection constructs are used (e.g., online shopping discounts, traffic light control). Write a short paragraph and create a simple flowchart for it.

4. **Debugging Exercise**:
   - Fix the following pseudocode:

```
INPUT temp
IF temp > 25 THEN
    OUTPUT "Warm"
ELSE IF temp > 15
    OUTPUT "Cool"
ELSE
    OUTPUT "Cold"
END
```

   Identify and correct syntax errors (e.g., missing THEN, END IF).

Additional Notes

- **Practical Tips**:
  - Use free tools like Draw.io or Lucidchart for flowchart creation, or draw by hand on paper.
  - Practice pseudocode in a text editor like Notepad++ to simulate coding without syntax errors.
- **Discussion Points**:
  - Why are flowcharts and pseudocodes useful before writing actual code?
  - How do selection constructs improve a program's flexibility?
- **Extension**:
  - For advanced students, introduce a simple programming language (e.g., Python) to implement one of the pseudocode examples (e.g., grading system) and compare it to the pseudocode.