

Teaching Plan for Sequence and Selection (If/Else) – G.C.E. (A/L) ICT

Audience

- Two G.C.E. (A/L) ICT students with no previous programming experience.
- Continuation of the Python introduction – variables, data types, operators and sequence were covered; selection control structures are taught here.

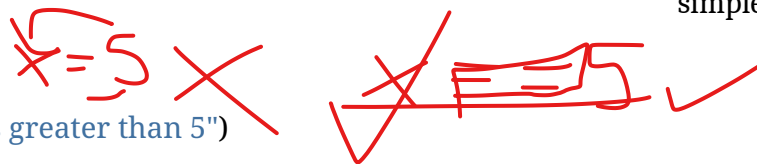
Objectives

- Understand the concept of *selection* (decision-making) in programming.
- Learn the syntax and usage of Python's if, if-else, if-elif-else and nested if statements.
- Recognise the role of indentation and logical conditions.
- Solve real-world problems by designing algorithms using sequence and selection.

Time Allocation (≈2 hours)

Time	Topic	Activities
0–10 min	Recap & Motivation	Recap previous lesson: variables, simple data types, arithmetic & logical operators, and sequence (straight-line execution). Introduce control structures: emphasise that programs can <i>make decisions</i> . Use the morning routine analogy (choose to carry an umbrella if it's raining) to illustrate selection in real life.
10–30 min	Simple if statement	Explain logical conditions and comparison operators (e.g., <code>a == b</code> , <code>></code> , <code><</code> , etc.) 【473172395057581†L862-L872】. Show syntax of a simple if statement:

```
x = 10
if x > 5:
    print("x is greater than 5")
```



Discuss indentation importance 【473172395057581†L895-L903】. **Exercise 1** (predict & test): Given `a = 7`, write code to print “Positive” only if `a` is greater than 0. **Exercise 2**: Given a fixed temperature (e.g., `temp = 30`), write code that prints “Hot day” when `temp` exceeds 28 °C. || **30–55 min | if-else statement** | Explain two-way selection: either execute one block or another. Demonstrate with example 【473172395057581†L941-L972】 :

```

age = 16
if age >= 18:
    print("You are an adult")
else:
    print("You are a minor")

```

Emphasise that the else clause runs when the condition is false. **Exercise 3:** Create a program that assigns num = 11 and prints “Even” if num is even, otherwise “Odd”. **Exercise 4:** Write code that checks whether a number score is at least 50 and prints “Pass” or “Fail” accordingly. | **55–80 min** | **if–elif–else chain** | Introduce multi-branch selection using elif (else-if) **[473172395057581†L920-L934]** . Explain that Python evaluates conditions in order and executes only the first true block. Example:

```

mark = 68
if mark >= 75:
    grade = "A"
elif mark >= 65:
    grade = "B"
elif mark >= 55:
    grade = "C"
else:
    grade = "F"
print("Grade:", grade)

```

Show how elif avoids deeply nested if statements. **Exercise 5:** Given n = 0, n = 7, and n = -3, modify code to print whether n is positive, negative or zero. **Exercise 6:** Program a simple tax calculator: if income ≤ 100,000 LKR, tax = 0; elif income ≤ 200,000 LKR, tax = 5 %; otherwise tax = 10 %. | **80–100 min** | **Nested if statements** | Explain that an if can contain another if inside it **[473172395057581†L1078-L1094]** . Use the *restaurant menu* analogy: choose a meal based on availability and preference. Example:

```

x = 15
if x > 0:
    print("Positive")
    if x % 2 == 0:
        print("and even")
    else:
        print("and odd")
else:
    print("Non-positive")

```

Discuss when nesting is appropriate (e.g., when multiple conditions depend on each other). **Exercise 7:** Given y = 20, write nested if logic that prints whether y is a multiple of 5 and/or 10. **Exercise 8:** Write a program that assigns age and gender and prints “Eligible for program” only if age ≥ 18 *and* gender is “F”; otherwise print appropriate messages. | **100–115 min** |

Shorthand if and conditional expressions | Show one-line if statements and ternary operators **【473172395057581†L978-L1005】** . Example: `print("A") if a > b else print("B")`. Discuss readability and recommend using full if statements for beginners. **Exercise 9:** Use a conditional expression to set `max_val` to the larger of two numbers `a` and `b`. **Exercise 10:** Create a one-line expression that prints “Yes” if a string contains a particular letter, else “No”. | **115–125 min** | **Combining Logical Operators in Conditions** | Demonstrate combining conditions with `and`, `or`, and `not` **【473172395057581†L1024-L1075】** . Example:

```
age = 20
citizen = True
if age >= 18 and citizen:
    print("Eligible to vote in Sri Lanka")
else:
    print("Not eligible to vote")
```

Exercise 11: Given fixed marks in ICT and Maths, write code that prints “Qualified for scholarship” only if both marks are ≥ 75 . **Exercise 12:** Check if either of two numbers is divisible by 3. | **125–135 min** | **Trace-through & Debugging Practice** | Demonstrate tracing code line by line (dry-run). Ask students to predict outputs before running programs. Use examples with nested conditions. Encourage them to walk through conditions, emphasising indentation and scope. | **135–145 min** | **Real-World Mini-Project** | Have students design and code a simple program using sequence and selection: e.g., *Electricity Bill Calculator* – given a number of units consumed, calculate the cost: first 64 units at Rs 5/unit, remaining units at Rs 10/unit. Walk them through algorithm design, variable assignment, conditions, and printing output. This question appeared in a past A/L ICT paper. | **145–150 min** | **Wrap-Up & Homework Assignment** | Recap the importance of selection and the difference between simple if, if-else, and if-elif-else. Highlight common pitfalls such as missing indentation or overlapping conditions. **Homework:** Assign three small tasks—(1) Write a program to determine if a year is leap year (selection logic). (2) Given a student’s age and average mark, decide if they qualify for different clubs (e.g., sports club ≥ 17 years and mark ≥ 60). (3) Convert 24-hour time to 12-hour format using selection. |

Teaching Techniques

- **Morning Routine Analogy** – Each step happens in sequence (wake up, brush teeth) but you might decide to take an umbrella or not based on rain (selection). Students relate to everyday decisions.
- **Menu Choice Analogy** – Use a restaurant menu to explain if-elif-else: if favourite dish is available choose it; else if second favourite is available choose that; else choose default. This helps them visualise multi-branch selection and nested decisions.
- **Dry-Run and Flowcharts** – Sketch flowcharts or draw simple diagrams illustrating the decision tree. Ask students to follow the “pointer” through conditions step by step, reinforcing the mental model of program execution.

Important Python Notes

- Logical conditions (equal to, greater than, etc.) are fundamental to if statements **【473172395057581†L862-L872】** .

if True: if False:

if $x > 0$ and $x < 10$

- Indentation defines the scope of the if block 【473172395057581†L895-L903】 . Misaligned indentation leads to errors.
 - elif allows checking multiple conditions in sequence 【473172395057581†L920-L934】 .
 - else captures all cases not handled by preceding if/elif blocks 【473172395057581†L941-L972】 .
 - Nested if statements enable multi-level decisions 【473172395057581†L1078-L1094】 .
 - Python also provides shorthand if expressions for simple assignments 【473172395057581†L978-L1005】 .
-

Appendix: Past Paper Questions and Model Questions

A. Past Paper-Style Questions on Sequence and Selection

The following questions are adapted from previous G.C.E. (A/L) ICT exams and revision papers. They cover sequence (straight-line execution) and selection (if/else). These examples do **not** require loops.

1. **Electricity Bill Calculation** – Past A/L Exam (2017):
“The consumption of electricity at houses is charged based on the number of units consumed. The first 64 units are charged at Rs 5.00 per unit; any additional unit is charged at Rs 10.00. Write a Python program that takes the number of units used (as a variable) and displays the total amount payable.” (Students should use if-else to test whether the consumption exceeds 64 units.)
2. **Previous and Current Meter Reading** – Past Paper:
“A household water meter records usage. The programme should store the previous meter reading and the current reading in variables, calculate the number of units used, and compute the payment using a fixed rate of Rs 15 per unit. Display the units and total payment.” (This question emphasises sequence for input/assignment and simple arithmetic; no condition is needed unless different rates are introduced.)
3. **Find the Maximum of Two Numbers** – Past Model Paper:
“Given two integer variables x and y, write a program that assigns max_val to the larger of the two numbers and prints it.” (Students can implement using if-else or the conditional expression.)
4. **Student Grading** – Past Revision Question:
“A student’s mark out of 100 is stored in a variable mark. Write code to assign a grade according to the following: $\text{mark} \geq 75$ – ‘A’, $60 \leq \text{mark} < 75$ – ‘B’, otherwise – ‘C’. Display the grade.” (Requires if-elif-else.)
5. **Classifying an Integer** – ICT Practice:
“Given a variable n, print whether n is positive, negative or zero.” (Requires a 3-way selection structure.)

$x = 0$

if x :

$x = 0$

elif:

6. **Tax Computation** – Past Paper Adaptation:

“The annual income of a person is stored in the variable income. Calculate the tax payable as follows: income up to 250,000 LKR → no tax; income between 250,001–500,000 LKR → 6 % of the income; above 500,000 LKR → 12 % of the income. Display the tax.” (*Encourages nested or chained if-elif-else.*)

B. Model MCQ Questions (Sequence & Selection)

1. **MCQ 1:** What will be printed by the following code?

```
a = 5
b = 10
c = 2
if b > a + c:
    print("A")
else:
    print("B")
print("C")
```

Options: A. A C

B. B C

C. A B

D. B only

Answer: A. Since $10 > 5 + 2$ is true, the first print statement prints A, and the subsequent print("C") (in sequence) prints C.

2. **MCQ 2:** Which of the following conditions correctly checks if the variable age is in the range 13–19 inclusive? A. if age \geq 13 and 19

B. if $13 \leq \text{age} \leq 19$

C. if age $>$ 13 or age $<$ 19

D. if (age \geq 13) or (age \leq 19)

Answer: B. Python allows chained comparisons like $13 \leq \text{age} \leq 19$, which returns True only when age is between 13 and 19 inclusive.

3. **MCQ 3:** What is the output of the following code?

```
x = -3
if x:
    print("Non-zero")
else:
    print("Zero")
```

A. Non-zero

B. Zero

C. Error

D. None

Answer: A. In Python any non-zero integer is considered *True*, so x evaluates to *True* and prints “Non-zero”.

4. **MCQ 4:** Consider:

```
a = 10
b = 5
if a > b:
    if a % b == 0:
        print("Divisible")
    else:
        print("Not divisible")
else:
    print("a is not greater than b")
```

What is printed? A. Divisible
B. Not divisible
C. a is not greater than b
D. None

Answer: A. $10 > 5$ is true and $10 \% 5 == 0$ is also true, so “Divisible” is printed.

5. **MCQ 5:** Which statement about if statements is **incorrect**? A. Indentation defines the block of code controlled by the if statement.
B. An else can be used without a preceding if.
C. You can have multiple elif branches after an if.
D. The colon (:) must follow the condition in an if statement.

Answer: B. An else must be part of an if/elif chain; it cannot stand on its own.

C. Model Essay-Type Questions (Sequence & Selection)

1. **Essay 1 – Billing Scenario:**

A telecommunication provider offers two packages. *Package A*: first 500 MB free; additional data at 0.02 USD per MB. *Package B*: flat rate of 20 USD per month for 1000 MB, plus 0.015 USD per MB beyond 1000 MB. A customer’s monthly usage and package type are stored in variables. Write a Python script (without using loops) that calculates the bill. Use selection to determine which rate applies, compute any extra charges, and display the final cost.

2. **Essay 2 – Marks and Rewards:**

A school awards students based on average marks in three subjects (ICT, Mathematics, English). If the average ≥ 75 , print “Distinction”; if $60 \leq \text{average} < 75$, print “Merit”; if $50 \leq \text{average} < 60$, print “Credit”; else print “Pass”. Write the program using if–elif–else statements. Show sample output for an average of 82 and 58.

3. **Essay 3 – Number Classification:**

Write a program that stores an integer num. The program should first determine whether num is positive, negative or zero. Then, inside the positive branch, use a nested if to check if it is even or odd and print appropriate messages. Explain how nested if statements help here.

4. **Essay 4 – Salary Increment:**

A company decides increments based on years of service and current basic salary. If service is ≥ 10 years, employees get a 15 % increment; if service is between 5–9 years, they get a 10 % increment; otherwise they get a 5 % increment. Write a Python program that calculates and prints the new salary based on variables `years_service` and `basic_salary`.

5. **Essay 5 – Admission Eligibility:**

An institution admits students into different programmes based on their stream and Z-Score. For example, if the stream is “Science” and $Z\text{-Score} \geq 1.5$, print “Eligible for Science”; elif the stream is “Commerce” and $Z\text{-Score} \geq 1.4$, print “Eligible for Commerce”; elif the stream is “Arts” and $Z\text{-Score} \geq 1.2$, print “Eligible for Arts”; else print “Not eligible”. Write the program using appropriate selection statements and suggest how to extend it for more streams.
