```
┌─────────────┐
│    Start    │
└──────┬──────┘
       │
       ▼
  ╱───────────╲
 ╱ Input 5 Test ╲
 ╲   scores    ╱
  ╲───────────╱
       │
       ▼
┌─────────────┐
│   Compute   │
└──────┬──────┘
       │
       ▼
  ╱───────────╲
 ╱    Print     ╲
 ╲   average   ╱
  ╲───────────╱
       │
       ▼
┌─────────────┐
│     End     │
└─────────────┘
```

Input Function

- The input function is **a built-in function** in Python that allows developers to <mark>read data from the user.</mark>

- The input function in python <u>reads the input as a string</u>, which can then be converted into other data types, such as integers, floating-point numbers, or Booleans

```python
name = input("Please Enter Your Name: ")
id= input("Please Enter Your Employee ID: ")
print("Name & Id: ", name, id)
```

```
>>> name = input("Please Enter Your Name: ")
    Please Enter Your Name: Ngu War Hlaing
>>> id= input("Please Enter Your Employee ID: ")
...
    Please Enter Your Employee ID: 12345
>>> print("Name & Id: ", name, id)
...
    Name & Id:  Ngu War Hlaing 12345
>>>
```

Example to input two float numbers and find their sum and average

```python
# python code to read two float numbers
# and find their addition, average

num1 = float(input("Enter first number : "))
num2 = float(input("Enter second number: "))

# addition
add = num1 + num2

# average
avg = add/2

print("addition: ", add)
print("average : ", avg)
```

Output

```
Enter first number : 123.456
Enter second number: 789.02
addition:  912.476
average :  456.238
```

Python round() Function

Example

Round a number to only two decimals:

```
x = round(5.76543, 2)
print(x)
```

5.77

How many decimal points you want

Example

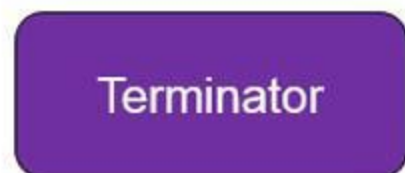Round to the nearest integer:

```
x = round(5.76543)
print(x)
```

6

Flowchart Symbols

Flowcharts are used to illustrate algorithms in order to aid in the visualisation of a program.

Flowcharts are to be read top to bottom and left to right in order to follow an algorithms logic from start to finish. Below is an outline of symbols used in flowcharts.

Terminator	**Terminator** Used to represent the Start and end of a program with the Keywords BEGIN and END.

Decision	**Decision** Used to split the flowchart sequence into multiple paths in order to represent SELECTION and REPETITION.

Process	**Process** An instruction that is to be carried out by the program.

Input / Output	**Input / Output** Used to represent data entry by a user or the display of data by the program.

→	**Arrow** Indicates the flow of the algorithm pathways.

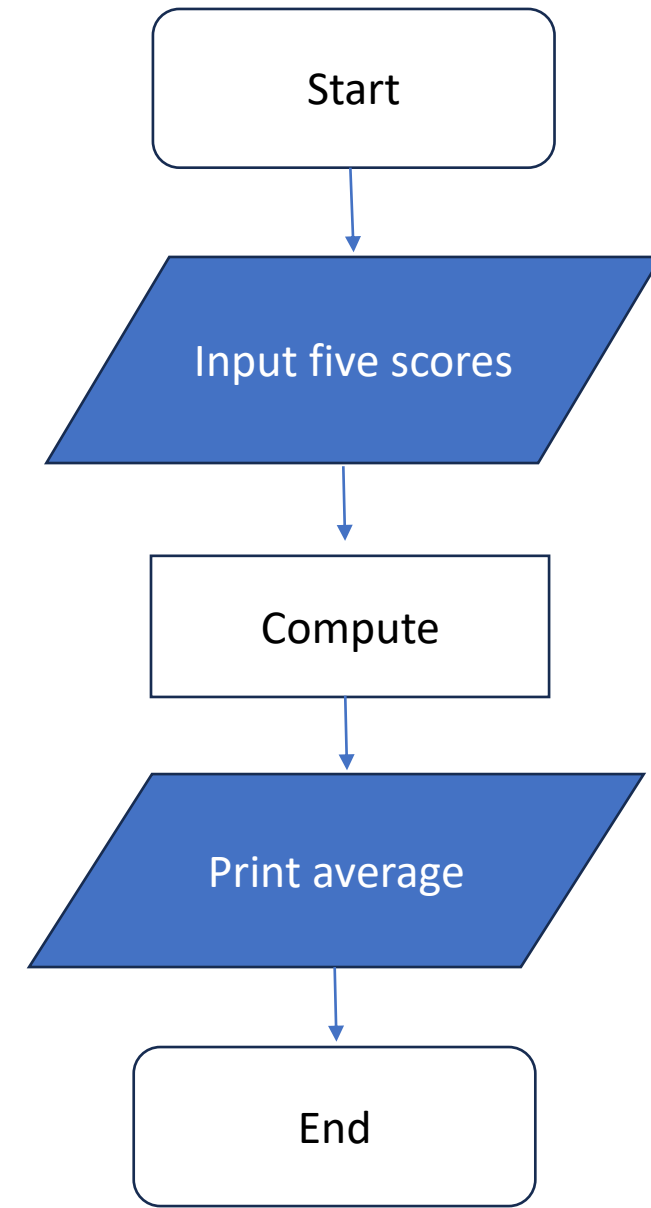|	
Subprogram	**Subprogram** References another program within the program.

Q1. Write a program that prompts the user to enter **five test scores** and then prints the **average of the test scores.**

1. Define variables with input function

2. Computational (average = total values/total numbers)

3. Print

Start

Input five scores

Compute

Print average

End

Q1. Write a program that prompts the user to enter **five test scores** and then prints the **average of the test scores.**

```python
# Question 1
userInput1 = float(input("Enter score 1: "))
userInput2 = float(input("Enter score 2: "))
userInput3 = float(input("Enter score 3: "))
userInput4 = float(input("Enter score 4: "))
userInput5 = float(input("Enter score 5: "))

score_total = (userInput1 + userInput2 + userInput3 + userInput4 + userInput5) / 5
print('Average score: ', round(score_total, 2))
```

Use the + operator

```
str1="Hello"
str2="World"
print ("String 1:",str1)
print ("String 2:",str2)
str=str1+str2
print("Concatenated two different strings:",str)
```

The + operator can be used to **concatenate two different strings.**

```
String 1: Hello
String 2: World
Concatenated two different strings: HelloWorld
```

**Q2.** Write a program that converts from **RM to SGD** (Singapore Dollar). On 1st May 2020, where 1 SGD is RM 3.04. Prompt the user to enter an RM amount and print the amount in SGD.

- Prompt User to enter amount of RM
- Calculation
- Print

```
# Question 2
myr_amount = float(input("Enter MYR value: MYR "))
print('The amount in SGD is : SGD' + str(round(myr_amount / 3.04,2)))
```

Start

Prompt user to enter amount in RM

Convert RM to SGD: RM / 3.04

Print RM amount in SGD

End
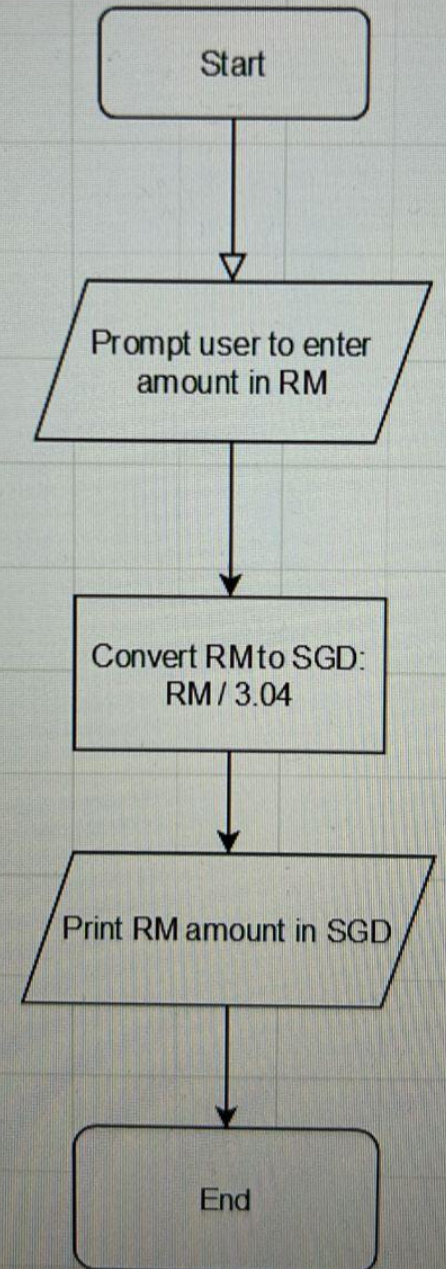
Q3. Write a program that **prompts the user to enter the length and height of a rectangle** and prints the area and perimeter of the rectangle.

- ▪ Define <u>length and height with input functions</u>

- ▪ Formulations for two equations (A=L*H)

- ▪ Print

PERIMETER
of a Rectangle
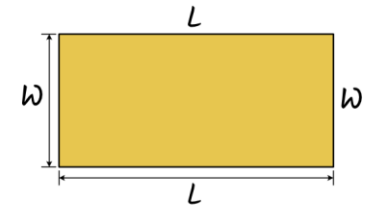
Formula:

$$P = 2L + 2W$$

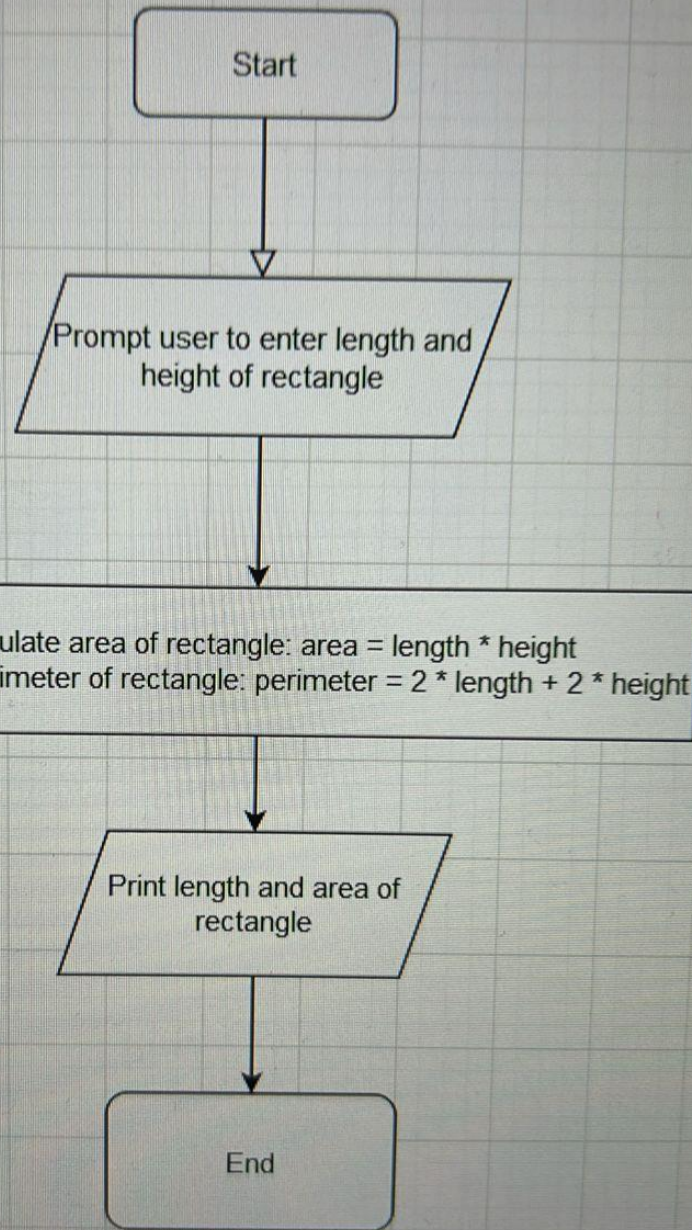where P stands for **Perimeter**, L for **Length** and W for **Width**

© chilimath.com

```
# Question 3
length = float(input("Enter the length of the rectangle: "))
height = float(input("Enter the height of the rectangle: "))
area = length * height
perimeter = (2 * length) + (2 * height)

print("The area of the rectangle is: " + str(area))
print("The perimeter of the rectangle is: " + str(perimeter))
```

Start

Prompt user to enter length and height of rectangle

Calculate area of rectangle: area = length * height
Calculate perimeter of rectangle: perimeter = 2 * length + 2 * height

Print length and area of rectangle

End

PERIMETER
of a Rectangle
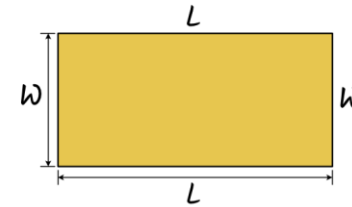
Formula:

$$P = 2L + 2W$$

where P stands for **Perimeter**, L for **Length** and W for **Width**

© chilimath.com

- Define <u>length and height with input functions</u>

- Formulations for two equations (A=L*H)

- Print

Q4. Write a program that accepts an object's mass (in kilograms) and velocity (in meters per second) as inputs and then outputs its momentum and kinetic energy given the following information.

- ▪ Define **mass** and **velocity** with input functions

- ▪ Formulations for two equations

- ▪ Print

```
# Question 4
mass = float(input("Enter the object's mass: "))
velocity = float(input("Enter the object's velocity: "))

momentum = round(mass * velocity, 2)
kineticEnergy = round((mass * velocity ** 2) / 2,2)

# Display the results
print("The object's momentum is " + str(momentum))
print("The object's kinetic energy is " + str(kineticEnergy))
```

Start

Prompt user to enter mass and velocity of object

Calculate momentum = mass * velocity
Calculate KineticEnergy = 0.5 * mass * velocity**2

Print momentum and KineticEnergy

End
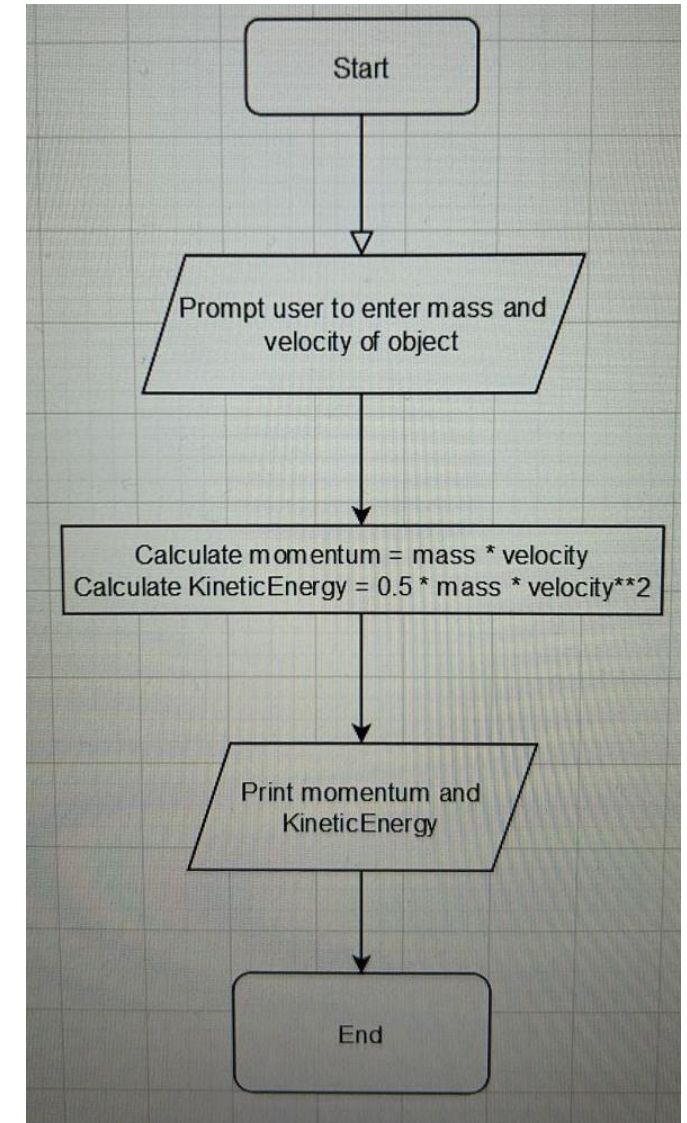
Q5. Write a program that prompts the user to input the elapsed time for an event in seconds. The program **then outputs the elapsed time in hours, minutes, and seconds.** For example, if the elapsed time is 9630 seconds, then the output would be 2:40:30.

Input: Enter elapsed time in seconds

Processing: Calculate hours, minutes, and seconds

1hr = 60 mins = 60*60 seconds = 3600 seconds

Output: Display elapsed time as hours:minutes:seconds

Start

Inputs:
Elapsed time
(seconds) as
int

Compute:
Calculate hour= seconds/3600

Remaing seconds= elapsed seconds % 3600

Minutes= Remaining seconds //60

Seconds= remaining seconds % 60

Print
hours:minute
s:seconds

Stop
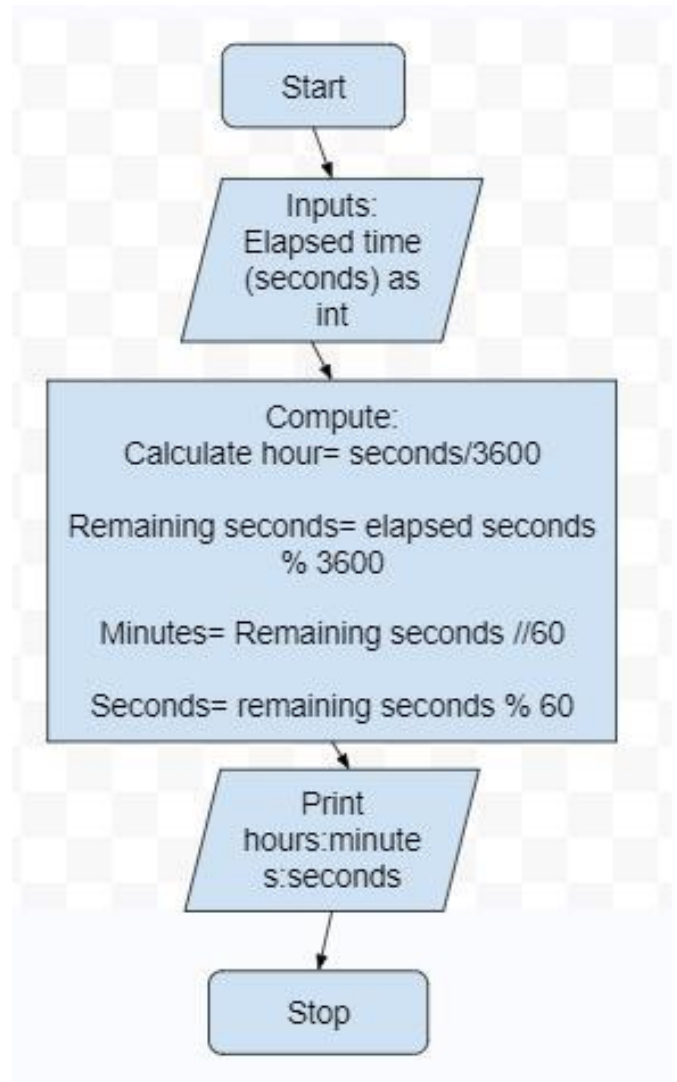
```python
# Prompt the user to input the elapsed time in seconds
elapsed_time_seconds = int(input("Enter the elapsed time in seconds: ")

# Calculate hours, minutes, and seconds
hours = elapsed_time_seconds // 3600
remaining_seconds = elapsed_time_seconds % 3600
minutes = remaining_seconds // 60
seconds = remaining_seconds % 60

# Print the result
print(f"The elapsed time is: {hours}:{minutes:02d}:{seconds:02d}")
```

Operation	Description	Type (if x and y are integers)
x + y	Sum	Integer
x - y	Difference	Integer
x * y	Product	Integer
x / y	Division	Float
x // y	Integer division	Integer
x % y	Remainder of x // y	Integer
-x	Negative x	Integer
abs(x)	Absolute value of x	Integer
x**y	x to the power y	Integer

Q6. Write a program that splits a restaurant bill among a group of friends. The program prompts the user to **input the total of the bill** and the **number of friends splitting the bill**. The program first calculates and adds a 10% service charge to the total, and then calculates and adds a 6% GST to the total (including the service charge). The program then outputs the amount to be paid by each friend.

```python
# Prompt the user to input the total bill and the number of friends
total_bill = float(input("Enter the total bill amount: "))
num_friends = int(input("Enter the number of friends splitting the bill

# Calculate service charge (10% of the total bill)
service_charge = 0.10 * total_bill

# Calculate GST (6% of the total bill including the service charge)
gst = 0.06 * (total_bill + service_charge)

# Calculate the total amount to be paid
total_amount = total_bill + service_charge + gst

# Calculate the amount to be paid by each friend
amount_per_friend = total_amount / num_friends

# Print the result
print(f"Each friend needs to pay: {amount_per_friend:.2f} MYR")
```

- Input Total bills and total friends

- Calculate 10% service charge, add 6% GST to the total, payment average

- Print
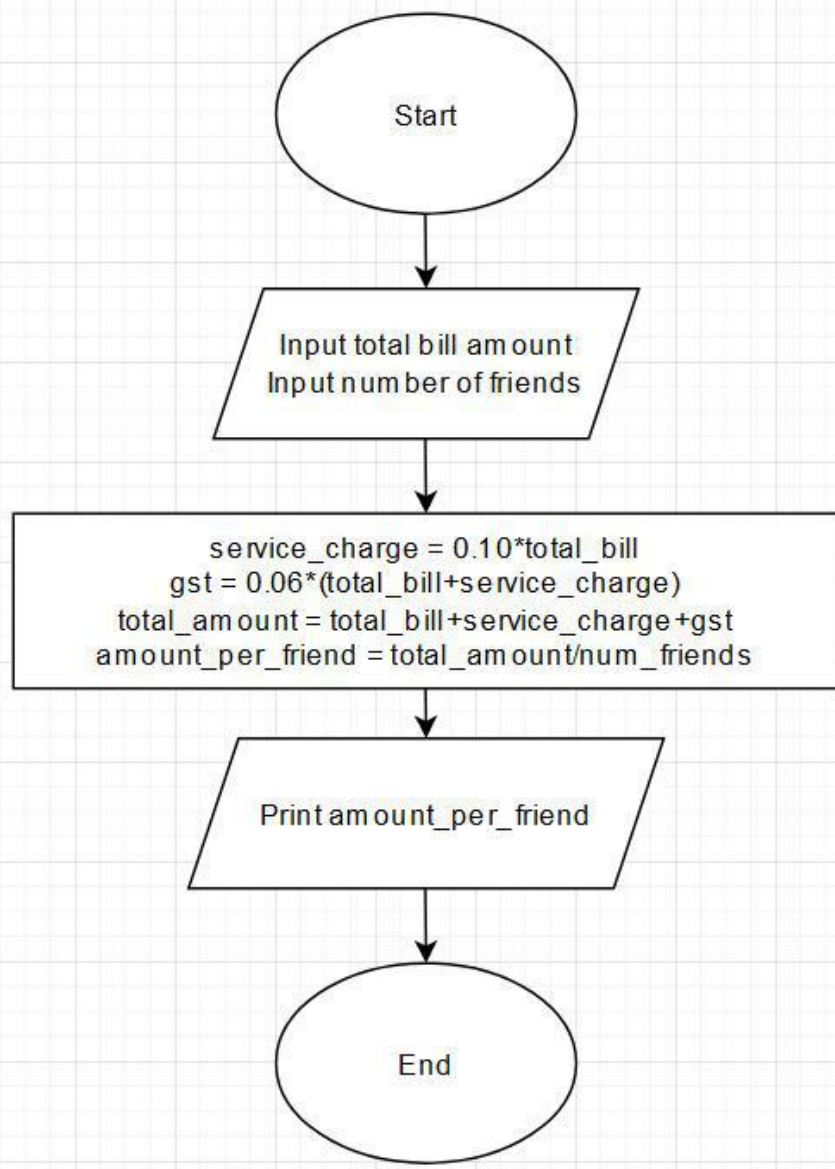
```
Start
```

```
Input total bill amount
Input number of friends
```

```
service_charge = 0.10*total_bill
gst = 0.06*(total_bill+service_charge)
total_amount = total_bill+service_charge+gst
amount_per_friend = total_amount/num_friends
```

```
Print amount_per_friend
```

```
End
```

```python
# Prompt the user to input the total bill and the number of friends
total_bill = float(input("Enter the total bill amount: "))
num_friends = int(input("Enter the number of friends splitting the bill

# Calculate service charge (10% of the total bill)
service_charge = 0.10 * total_bill

# Calculate GST (6% of the total bill including the service charge)
gst = 0.06 * (total_bill + service_charge)

# Calculate the total amount to be paid
total_amount = total_bill + service_charge + gst

# Calculate the amount to be paid by each friend
amount_per_friend = total_amount / num_friends

# Print the result
print(f"Each friend needs to pay: {amount_per_friend:.2f} MYR")
```

if...else Statement

The `if...else` statement is used to execute a block of code among two alternatives.

However, if we need to make a choice between more than two alternatives, we use the `if...elif...else` statement.

Example: Python if...elif...else Statement

```python
number = 10

if number > 0:
    print('Positive number')

else:
    print('Negative number')

print('This statement always executes')
```

```python
number = 0

if number > 0:
    print('Positive number')

elif number <0:
    print('Negative number')

else:
    print('Zero')

print('This statement is always executed')
```

```
Positive number
This statement always executes
```

Output

```
Zero
This statement is always executed
```

Q7. Write a program that prompts the user for his/her yearly income, and outputs the amount of taxes to pay based on the yearly income. The tax table is as follows:

- RM 0 to 2,500 – Tax rate: 0%
- RM 2,501 to 10,000 – Tax rate: 5%
- RM 10,001 to 50,000 – Tax rate: 15%
- Exceeding 50,001 – Tax rate: 25%

For example, if the income is RM 12,000, the total taxes would be RM 1,800 (15%).

```
# Question 7
yearly_income = float(input('Enter yearly income in RM: '))

if yearly_income >= 0.0 and yearly_income <= 2500.0:
    total_taxes = 0.0
elif yearly_income <= 10000.0:
    total_taxes = yearly_income * (5 / 100)
elif yearly_income <= 50000.0:
    total_taxes = yearly_income * (15 / 100)
else:
    total_taxes = yearly_income * (25 / 100)

print('The total taxes for yearly income RM' + str(yearly_income) + ' is RM' + str (total_taxes) )
```

```
Start
  │
  ▼
Prompt user to enter
  yearly_income
  │
  ▼
Is yearly_income >= 0      ──Yes──►   Calculate tax =
and <=2500?                            yearly_income * 0
  │
  No
  │
  ▼
Is yearly_income <=        ──Yes──►   Calculate tax =
10000?                                 yearly_income * 0.05
  │
  No
  │
  ▼
Is yearly_income <=        ──Yes──►   Calculate tax =
50000?                                 yearly_income * 0.15
  │
  No
  │
  ▼
Calculate tax =           ──►   Print yearly_income   ──►   End
yearly_income * 0.25            and tax
```
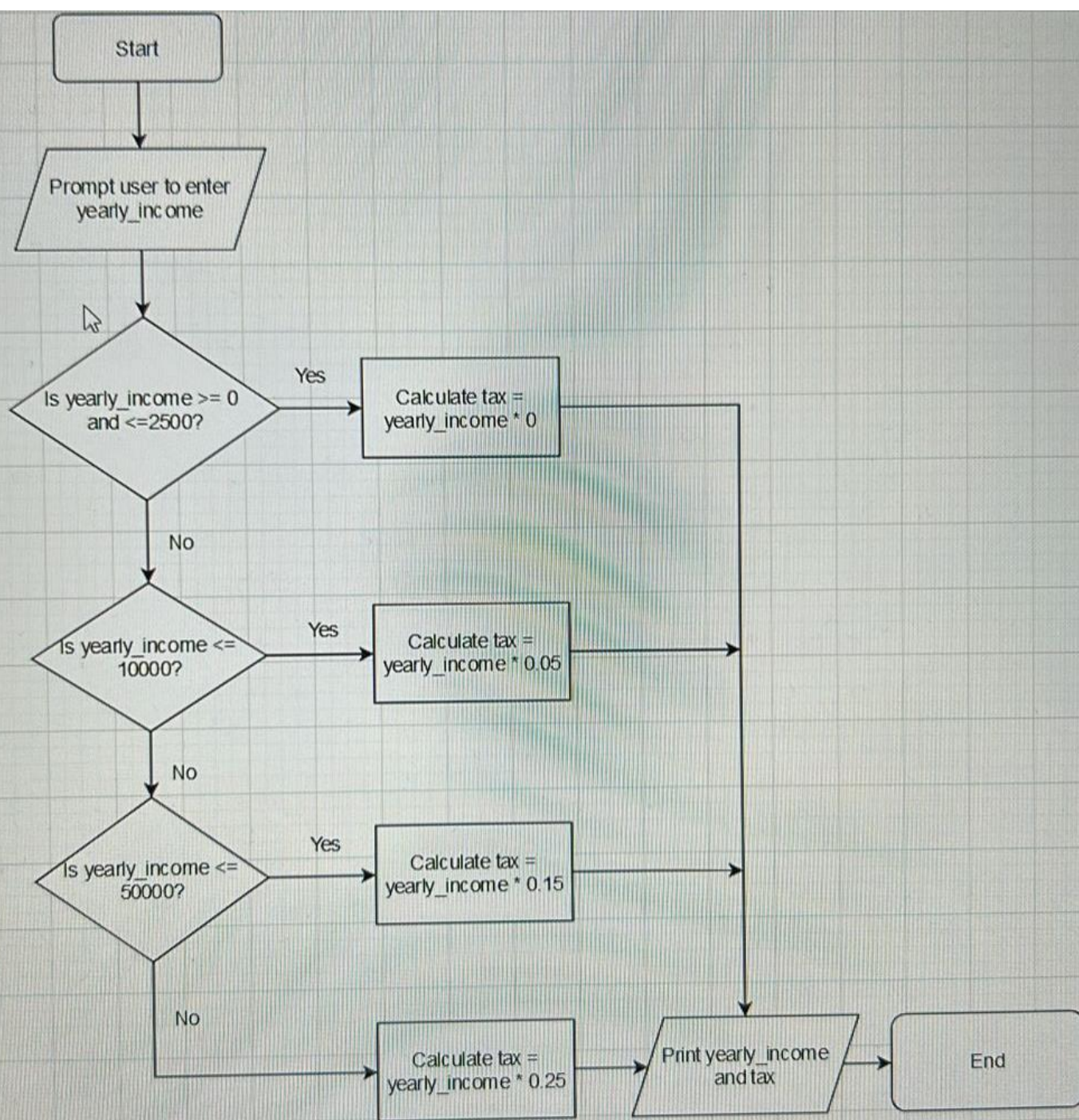
Q8. Company ABC is introducing a new data plan for smartphones. **Each GB (Gigabyte) of data will cost RM 15, up to 10 GB. Any data over 10 GB will be charged at RM 30 per GB.** Write a program that prompts the user to enter their monthly data usage (in GB) and prints the data charges for the month.

Two Conditions – If else

- If you are using equal or less than 10GB, you will be charged for RM 15 for each GB
- If more than 10 GB, any data after 10GB will be charged RM 30 for each.

```
# Question 8
data_usage = float(input('Enter data usage in GB: '))
if data_usage <= 10:
    data_charges = data_usage * 15

else:
    data_charges_10GB = 10 * 15
    data_charges_extra = (data_usage - 10) * 30
    data_charges = data_charges_10GB + data_charges_extra
print('The montly data charges for ' + str(data_usage) + 'GB is RM' + str(data_charges))
```

```
                    ┌─────────────┐
                    │    start    │
                    └─────────────┘
                           │
                           ▼
                    ╱──────────────╲
                   ╱ input monthly  ╲
                   ╲  data usage(GB) ╱
                    ╲──────────────╱
                           │
                           ▼
                        ╱─────╲              Yes   ┌────────────────────────────┐
                       ╱       ╲──────────────────▶│ calculate data charges:    │
                      ╱ dataUsage╲                  │ dataCharge = dataUsage * 15│
                      ╲ <= 10?  ╱                   └────────────────────────────┘
                       ╲       ╱                                │
                        ╲─────╱                                 │
                           │                                    │
                          No                                    │
                           │                                    │
                           ▼                                    │
        ┌─────────────────────────────────────┐                │
        │ calculate data charges:              │                │
        │ dataCharge = ((dataUsage - 10)*30)+150│               │
        └─────────────────────────────────────┘                │
                           │                                    │
                           ▼                                    │
                    ╱──────────────╲                            │
                   ╱ display         ╲◀─────────────────────────┘
                   ╲  dataCharge     ╱
                    ╲──────────────╱
                           │
                           ▼
                    ┌─────────────┐
                    │     end     │
                    └─────────────┘
```