Flow Chart, Pseudo Code
Data Type and Variable

Discussion 1

The following Pseudocode read in 10 students' scores and calculate the average.

```
Initialize student_counter to zero

While student_counter is less than or equal to ten

    Input the next score
    Add the score into the total

EndWhile

Set the class average to the total divided by ten
```

There are some errors in the above Pseudocode. Please indicate where the errors are and how to correct them.

- Initialize total
- Add student_counter near end of loop
- Initialize student_counter to one

Flow Chart, Pseudo Code
Data Type and Variable

Discussion 2

Fill in the blanks to complete the following Pseudocode to read in 10 students' scores and calculate the number of passes and failures.

```
Initialize passes to zero

Initialize failures to zero

Initialize student_counter to one

While student_counter is less than or equal to ten

    Input the next score

    If student passed
        add 1 to passes

    else if student failed
        add 1 to failure

    endif

    add one to student_counter

EndWhile

    print number of passes
    print number of failures
```

Flow Chart, Pseudo Code
Data Type and Variable

Discussion 3

Write the FizzBuzz algorithm using pseudocode.

FizzBuzz is a standard interview problem. The Problem state:

- Write a code that prints each number from 1 to 20 on a new line.
- Print “Fizz” if the number is the multiple of 3.
- Print “Buzz” if the number is multiple of 5
- For number which is multiple of both 3 and 5 print “FizzBuzz”

The ***sample run*** is as follows:

```
1
2
Fizz
4
Buzz
Fizz
7
8
Fizz
Buzz
11
Fizz
13
14
FizzBuzz
16
17
Fizz
19
Buzz
```

```
num_c = 1

while num_c <= 20:
   if num_c % 15 == 0:
      print("FizzBuzz")
   elif num_c % 3 == 0:
      print("Fizz")
   elif num_c % 5 == 0:
      print("Buzz")
   else:
      print(num_c)
   num_c += 1
```

Flow Chart, Pseudo Code
Data Type and Variable

Discussion 4

For each of the following, discuss what the outcome will be if they are executed by a Python interpreter (e.g.IDLE3) in the sequence shown.

```
c = 10                                  t
7 = a                                   f
a = d                                   f
a = c + 1                               t
a + c = c                               f
3 + a                                   t
7up = 10                                f
import = 1003                           f
b = math.pi * c                         t

                                        f
int = 500

                                        t
a ** 3                                  t
a,b,c = c,1,a                           f
b,c,a = a,b                             t
c = b = a = 7                           t
print( A )                              f
print( "b*b + a*a = c*c" )              t
print( 'A' )                           f
print( "c" = 1 )
```

Flow Chart, Pseudo Code
Data Type and Variable

Discussion 4

Write a program that asks the user for the number of boys and that of girls in a class. The program should calculate and display the percentage of boys and girls in the class. A sample run is as follows:

Enter the number of boys: 65

Enter the number of girls: 77

Boys: 46%

Girls: 54%

```
num_b = int(input("Enter the number of boys: "))
num_g = int(input("Enter the number of girls: "))
total = num_b + num_g

print("Boys: ",str( round(int(num_b) / total * 100)) + "%"
)
print("Girls: ", str( round(int(num_g) / total * 100)) + "%"
)
```

a) Design the algorithm and use flowchart to present.
b) Write the Python program.

Guideline for instructor

Discussion 1

The following Pseudocode read in 1o students' scores and calculate the average.

Initialize total to zero (Must initialize variable before use)

Initialize student_counter to one (if counter starts from zero, will be 11 students instead of 10)

While student_counter is less than or equal to ten

    Input the next score
    Add the score into the total

    Add one to student_counter (update looping control variable)

EndWhile

Set the class average to the total divided by ten

Discussion 2

```
Initialize passes to zero

Initialize failures to zero

Initialize student_counter to one

While student_counter is less than or equal to ten

     Input the next score
     If the student passed
          Add one to passes

     else

          Add one to failures

     EndIf

     add one to student counter

EndWhile

print the number of passes

print the number of failures
```

Flow Chart, Pseudo Code
Data Type and Variable

Discussion 3

```
num : 1

FOR num -> 1 to 20

    IF num MOD 15 ===0

        PRINT "FizzBuzz"

    ELSE IF num MOD 3 ===0

        PRINT "Fizz"

    ELSE IF num MOD 5===0

        PRINT "Buzz"

    ELSE

        PRINT num

    ENDIF

ENDFOR
```

Discussion 4

```
c = 10
```

valid, Python statement
Discussion:
What does the '=' symbol means? It is not the mathematic 'equal', but 'assignment' in programming.
This statement is to assign a value of 10 to a variable c

What is the data type?
The data type of C is integer.

Flow Chart, Pseudo Code
Data Type and Variable

What is a variable? A name created in a program that can be assigned a value, which can be stored, and modified if needed during the execution of the code.
Contrast this to a Literal (e.g. 123), which has a fixed value and cannot be changed.

Is this a good name to be used for a variable?
How do you choose a good name for a variable? We should use meaningful name to make our code more readable.
E.g. pi, which is most likely related to ? (mathematic π)

What is the advantage of using a variable? consider

```
pi = 3.1415926536
radius = 2.356
```

When we see this 2*pi*radius, we know it is to find the circumference
Compare with 2*3.1415926536*2.356 – error prone and not readable

```
7 = a
```

Invalid, LHS cannot be a literal

```
a = d
```

invalid, because d hasn't been created/defined

```
a = c + 1
```

valid, Python statement (LHS is a variable, RHS is a expression)

```
a + c = c
```

invalid, because LHS cannot be an expression, should be a variable

```
3 + a
```

valid, Python expression

Discussion: the difference between a statement and an expression?
Expression – return a value (when an expression is entered in the IDLE interpreter, the value is immediately printed on the following line)
Statement – does not return a value (when a statement is entered in the IDLE interpreter, nothing is shown on the following line)

```
7up = 10
```

invalid, Python naming convention: cannot start a variable name with digits

must start with a leter or underscore character (_);

```
import = 1003
```

invalid, import is a reserved word (known as keyword) in Python

Discusion: Usage - *import module*
module = a file that contains definition of variables and python code (functions) ans statements, which can be import into your program code and execute.
e.g. *import math* (which contains the definition of pi)

```
b = math.pi * c
```

invalid, we need to first "import math" for math module

```
int = 500
```

valid, int is not a keyword in python, although it can be used to cast non-integer data type to integer. Not recommend to be used as name of variable to avoid ambiguity. (similarly for float, input etc)

```
a ** 3
```

valid, Python expression- ** is an arithmetic operator for exponential

```
a,b,c = c,1,a
```

valid, Python statement for multiple assignment

```
b,c,a = a,b
```

invalid, different number of elements on the two sides: LHS and RHS

```
c = b = a = 7
```

valid, Python allows chained assignment even though b is just created in the middle

```
print( A )
```

invalid, A not defined (Python: case sensitive)

```
print( "b*b + a*a = c*c" )
```

valid, Python statement

```
print( 'A' )
```

valid, Python treats single quotes to be the same as double quotes

```
print( "c" = 1 )
                invalid, because we cannot assign 1 to c and then do
                print in a single Python statement
```

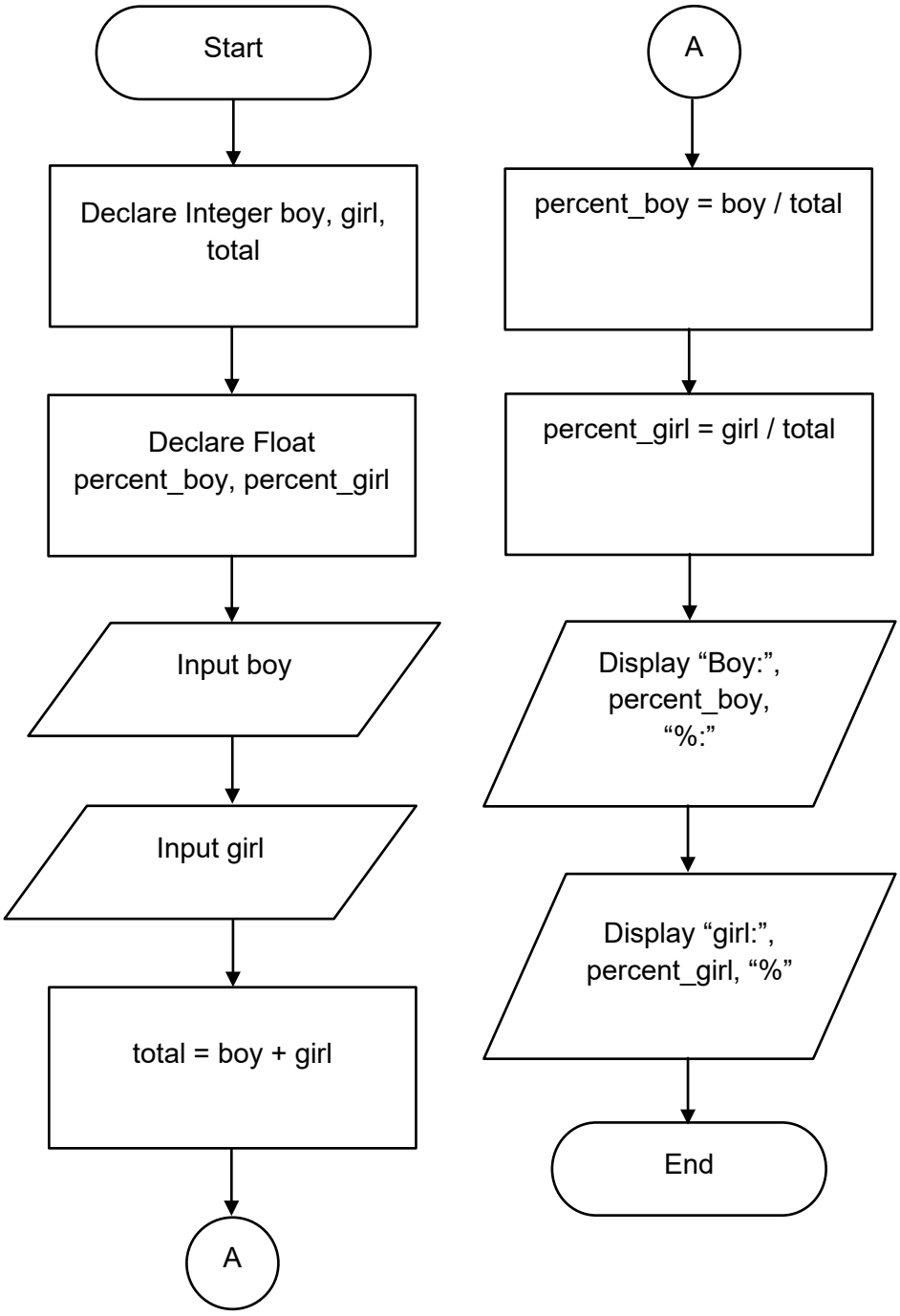
Discussion 5

- Flowchart is more effective in term of visualization, such as to observe easily whether all decision points are covered.
- It is easy to use during the initial design stage, such as sketching it on paper or on whiteboard for discussion.
- It may be quite tedious to draw using computer unless special software tool are use (Modern tool can also generate flowchart from program code and vice versa)
- As the size of the algorithm increases, flowchart may become too complicated to be drawn (This is where the concept of abstraction will be useful, as will be learnt later)

- Pseudo code is closer to the way we code a program, and hence can be easily translated to the specific program language.
- But it will be more time consuming to produce

In practice, start with flowchart, then use pseudocode before translating them to the program statements in specific language

Flow Chart, Pseudo Code
Data Type and Variable



```
# Get the number of boys.
boys = int(input("Enter the number of boys: "))

# Get the number of girls.
girls = int(input("Enter the number of girls: "))

# Calculate the total number of students.
total = boys + girls


##output version 1: round function
percent_boys = round(boys/total*100)
percent_girls = round(girls/total*100)

# Print the percentage of boys.
print("Boys:",str(percent_boys) + "%")
# Print the percentage of boys.
print("Girls:",str(percent_girls) + "%")

##output version 2: format function using %
# Calculate the percentage of boys.
percent_boys = boys / total

# Calculate the percentage of girls.
percent_girls = girls / total
# Print the percentage of boys.
print("Boys:", format(percent_boys, ".0%"))

# Print the percentage of girls.
print("Girls:", format(percent_girls, ".0%"))
```