SFU SIMON FRASER UNIVERSITY
ENGAGING THE WORLD

CMPT 120: Introduction to Computing Science and Programming 1

Procedural programming in Python

python™

One-Stop Access To Course Information

- **Course website**: One-stop access to all course information.

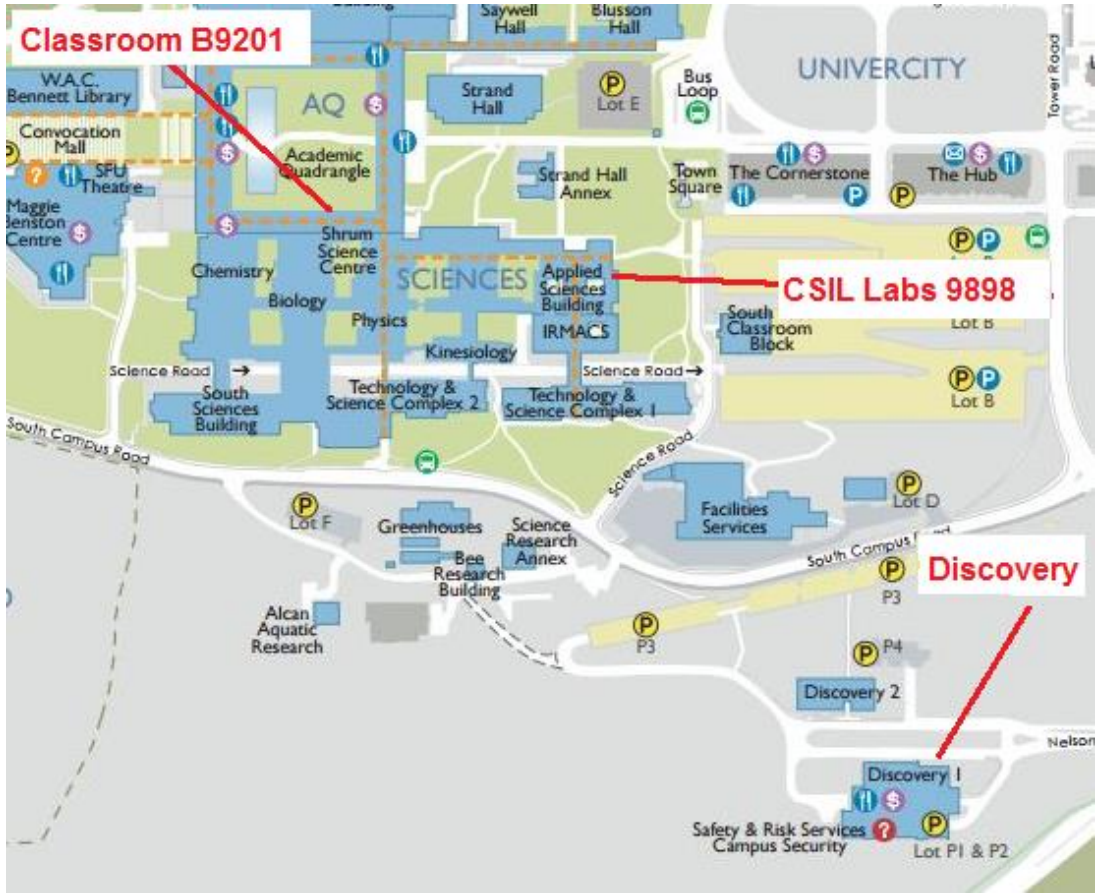
  http://www2.cs.sfu.ca/CourseCentral/120/liaqata/WebSite/index.html

- Course Outline
- Exam Schedule
- Python Info
- CourSys/Canvas link
- Learning Outcomes
- Office Hours
- Textbook links
- and more…
- Grading Scheme
- Lab/Tutorial Info
- Assignments

- **Canvas**: Discussions forum - https://canvas.sfu.ca/courses/39187
- **CourSys**: Assignments submission, grades - www.coursys.sfu.ca

Some Reminders

- Get familiar with the course Website.
  - http://www2.cs.sfu.ca/CourseCentral/120/liaqata/WebSite/index.html
  - Minor updates may occur during first week.
- Get fob to access LABS **(start next week!)**
  - If you don't have it already, get a new fob from Discovery Park 1 .



Additional Resources / Online References

- Online references are **as important as the texts**. (Links on course website.)
- These resources are **very important to your success**.
  - They aren't meant to be read from beginning to end like the readings in the textbook.
- You should **use them to get an overall picture of the topic** and as references as you do the assignments.

How to Learn in This Course?

A **Attend** Lectures & Labs

R **Read** / review Textbook/Slides/Notes

R **Reflect** and ask Questions

O **Organize** – your learning activities on weekly basis,

and finally…

W **Write** Code, Write Code, and Write Code.

Course Topics

1. General introduction
2. Algorithms, flow charts and pseudocode
3. **Procedural programming in Python**
4. Data types and control structures
5. Fundamental algorithms
6. Binary encodings
7. Basics of computability and complexity
8. Basics of Recursion
9. Subject to time availability:
   - Basics of Data File management

Today's Topics

1. Procedural Programming in Python

- **Transform an algorithm to a program:**
- **Write** Code, Write Code, and Write Code.

Today's Topics

1

Transform an algorithm to a program:

Write Code, Write Code, and Write Code.

Write a Python Program to Add Two Numbers

Step 1: Start

Step 2: Declare a variable n1 and initialize it to 0.

Step 3: Declare a variable n2 and initialize it to 0.

Step 4: Declare a variable sum and initialize it to 0.

Step 5: Get 1st number from user and store in n1.

Step 6: Get 2nd number from user and store in n2.

Step 7: Add n1 and n2, and store the answer in sum.

Step 8: Display SUM

Step 9: End

```
n1 = 0
n2 = 0
sum = 0
n1 = input()
n2 = input()
sum = n1 + n2
print(sum)
```

Write a Python Program: Add Two Numbers

Step 1: Start

Step 2: Declare a variable n1 and initialize it to 0.

Step 3: Declare a variable n2 and initialize it to 0.

Step 4: Declare a variable sum and initialize it to 0.

Step 5: Get 1st number from user and store in n1.

Step 6: Get 2nd number from user and store in n2.

Step 7: Add n1 and n2, and store the answer in sum.

Step 8: Display SUM

Step 9: End

```
n1 = 0
n2 = 0
sum = 0
n1 = input()
n2 = input()
sum = int(n1)+int(n2)
print(sum)
```

Write a Python Program to Add Three Numbers

Step 1: Start

Step 2: Declare a variable n1 and initialize it to 0.

Step 3: Declare a variable n2 and initialize it to 0.

Step 4: Declare a variable n3 and initialize it to 0.

Step 5: Declare a variable sum and initialize it to 0.

Step 6: Get 1st number from user and store in n1.

Step 7: Get 2nd number from user and store in n2.

Step 8: Get 3rd number from user and store in n3.

Step 9: Add N1 and N2 and assign the result to SUM.

SUM ← N1 + N2

Step 10: Display SUM

Step 11: End

```
n1 = 0
n2 = 0
n3 = 0
sum = 0
n1 = input()
n2 = input()
n3 = input()
sum = int(n1) + int(n2)+int(n3)
print(sum)
```

Write a Python Program to Solve 2x+2y

Step 1: Start

Step 2: Declare a variable x and initialize it to 0.

Step 3: Declare a variable y and initialize it to 0.

Step 4: Declare a variable ans and initialize it to 0.

Step 5: Get value of x from user and store in x.

Step 6: Get value of y from user and store in y.

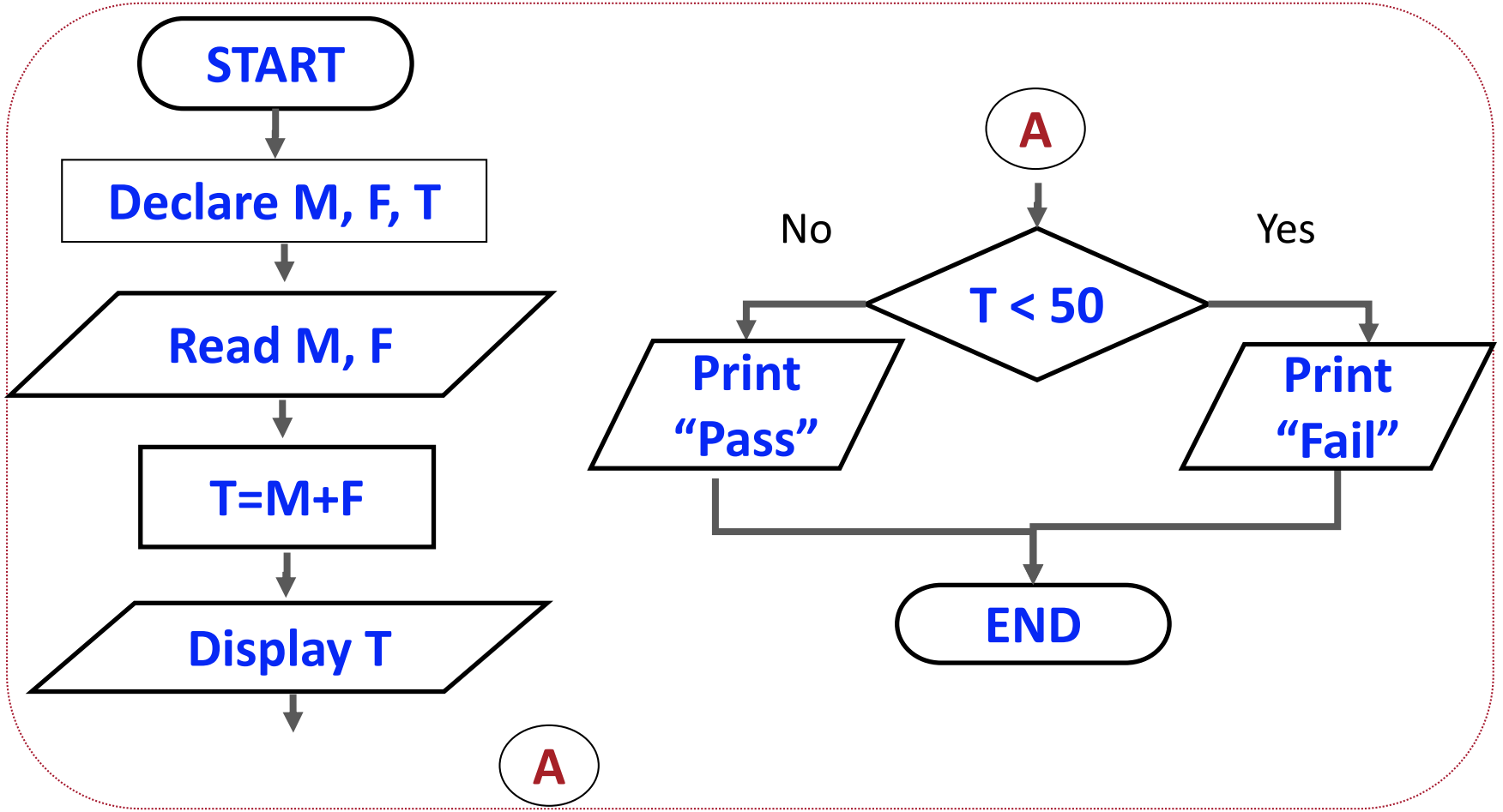
Step 7: Solve the expression: 2*x + 2 * y.

Step 8: Display SUM

Step 9: End

```
x = 0
y = 0
ans = 0
x = input()
y = input()
ans = 2*int(x) + 2*int(y)
print(ans)
```

Write a Program for Following Marks Flowchart




```
M = 0
F = 0
T = 0
M = input()
F = input()
T = int(M) + int(F)
print(T)
if (T<50): print("fail")
else: print("pass")
```

Program: Find the Smaller of Two Numbers

Step 1: Start

Step 2: Declare variable n1 to store the 1st number.

Step 3: Declare variable n2 to store the 2nd number.

Step 4: Get the value of n1 from the user.

Step 5: Get the value of n2 from the user.

Step 6: If n1 < n2 then print n1.
else print n2.

Step 8: End

```
n1 = 0
n2 = 0
n1 = input()
n2 = input()
if (n1 < n2):
   print(n1)
else:
  print(n2)
```

Program: Find the Smallest of Three Numbers

Read n1, n2, n3.

If n1 < n2 and n1 < n3,
Write n1.

If n2 < n1 and n2 < n3,
Write n2.

If n3 < n1 and n3 < n2,
Write n3.

```
n1 = 0
n2 = 0
n3 = 0
n1 = input()
n2 = input()
n3 = input()
if (n1 < n2) and : (n1 < n3) :   print(n1)
if (n2 < n1) and : (n2 < n3) :   print(n2)
if (n3 < n1) and : (n3 < n2) :   print(n3)
```

Program: Find Sum of First 100 Natural Numbers

Set sum to 0

Set n to 1

Repeat until n <=100:

  Set sum = sum + n

  n = n + 1

Write S

```
sum = 0
n = 1
while (N <=100):
  sum = sum + n
  n = n + 1
print(sum)
```

Algorithm: Convert Height In Meters To Feet and Inches

1: Start
2: Declare meter, feet, total inches and inches variables.
3: Initialize feet, total inches and inches variables to 0.
4: Get the height in meters from the user.
5: Convert meters into total inches and store it.
5: Convert total inches into feet and store it.
6: Find remainder of total inches / 12 and store in inches.
7: Display the value in feet variable.
8: Display the value in the inches variable.
9: End

Read *meters*

Set *totInch to 39.37 × metres*

Set *feet to whole number part of totInch / 12*

Set *inches to remainder of totInch / 12*

Write *feet, inches*

Program: Convert Height In Meters To Feet and Inches

Read *meters*

Set *totIal_inch to 39.37 × metres*

Set *feet to whole number part of totInch / 12*

Set *inches to remainder of totInch / 12*

Write *feet, inches*

Submit on Canvas today by midnight

```
print("Convert height in meters to feet and inches.")
meters = 0
feet = 0
inches = 0
total_inch = 0
meters = input("Enter height in meters: ")
meters = float(meters)
total_inch = meters * 39.37;
feet = total_inch // 12
inches = total_inch % 12
print("Height is", feet,"feet and",inches, "inches")
```

Questions?