COL1000
Introduction to Programming

Priyanka Golia

Most (if not all) of the content is borrowed from Prof. Subodh Kumar's slides

Mid Term!

- Web user security
- Variables, constants, and types
- Assignment (with unpacking and order of evaluation), Expressions (arithmetic, relational, logical, conditional, including short-circuiting and operator precedence)
- Conditions
- Control flow (including conditional statement, loops, and nesting)
- Lists (including lists of lists and list comprehension),
- Tuples
- Strings
- Error handling and comprehension
- Objects and references
- Mutability and Immutability.
- **The weightage of the midterm exam is 29%.**

Remember to go through each instructor's slides. Practice, practice, and practice!

Quiz

```
a = b = c = [1, 2]
a.append(3)
print(b, c)
```

```
a, b = [1, 2]
print(a,b)
```

```
a, b = [1, 2,3]
print(a,b)
```

```
a, *b= [1, 2,3]
print(a,b)
```

```
y[2], y = "abc", "z"
```

```
s = [(i, j) if i < j for i in [1, 2, 3] for j in [1, 2, 3]]
print(s)
```

```
s = "banana"
print(s.replace("na", "xy", 1))
```

```
s = "a,b,c,d"
print(s.split(",", 2))
```

```
s = "a,b,c,d"
print(s.split(",", -1))
```

```
s = "a,b,c,d"
print(s.split(",", 5))
```

Quiz

```
a = b = c = [1, 2]
a.append(3)
print(b, c)
```

[1,2,3] [1,2,3]

```
a, b = [1, 2]
print(a,b)
```

```
a, b = [1, 2,3]
print(a,b)
```

```
a, *b= [1, 2,3]
print(a,b)
```

```
y[2], y = "abc", "z"
```

```
s = [(i, j) if i < j for i in [1, 2, 3] for j in [1, 2, 3]]
print(s)
```

```
s = "banana"
print(s.replace("na", "xy", 1))
```

```
s = "a,b,c,d"
print(s.split(",", 2))
```

```
s = "a,b,c,d"
print(s.split(",", -1))
```

```
s = "a,b,c,d"
print(s.split(",", 5))
```

Quiz

```
a = b = c = [1, 2]
a.append(3)
print(b, c)
```

[1,2,3] [1,2,3]

```
a, b = [1, 2]
print(a,b)
```

1 2

```
a, b = [1, 2,3]
print(a,b)
```

```
a, *b= [1, 2,3]
print(a,b)
```

```
y[2], y = "abc", "z"
```

```
s = [(i, j) if i < j for i in [1, 2, 3] for j in [1, 2, 3]]
print(s)
```

```
s = "banana"
print(s.replace("na", "xy", 1))
```

```
s = "a,b,c,d"
print(s.split(",", 2))
```

```
s = "a,b,c,d"
print(s.split(",", -1))
```

```
s = "a,b,c,d"
print(s.split(",", 5))
```

Quiz

```
a = b = c = [1, 2]
a.append(3)
print(b, c)
```

[1,2,3] [1,2,3]

```
a, b = [1, 2]
print(a,b)
```

1 2

```
a, b = [1, 2,3]
print(a,b)
```

ValueError: too many values to unpack (expected 2)

```
a, *b= [1, 2,3]
print(a,b)
```

```
y[2], y = "abc", "z"
```

```
s = [(i, j) if i < j for i in [1, 2, 3] for j in [1, 2, 3]]
print(s)
```

```
s = "banana"
print(s.replace("na", "xy", 1))
```

```
s = "a,b,c,d"
print(s.split(",", 2))
```

```
s = "a,b,c,d"
print(s.split(",", -1))
```

```
s = "a,b,c,d"
print(s.split(",", 5))
```

Quiz

```
a = b = c = [1, 2]
a.append(3)
print(b, c)
```

[1,2,3] [1,2,3]

```
a, b = [1, 2]
print(a,b)
```

1 2

```
a, b = [1, 2,3]
print(a,b)
```

ValueError: too many values to unpack (expected 2)

```
a, *b= [1, 2,3]
print(a,b)
```

1 [2,3]

```
y[2], y = "abc", "z"
```

```
s = [(i, j) if i < j for i in [1, 2, 3] for j in [1, 2, 3]]
print(s)
```

```
s = "banana"
print(s.replace("na", "xy", 1))
```

```
s = "a,b,c,d"
print(s.split(",", 2))
```

```
s = "a,b,c,d"
print(s.split(",", -1))
```

```
s = "a,b,c,d"
print(s.split(",", 5))
```

Quiz

```
a = b = c = [1, 2]
a.append(3)
print(b, c)
```

[1,2,3] [1,2,3]

```
a, b = [1, 2]
print(a,b)
```

1 2

```
a, b = [1, 2,3]
print(a,b)
```

ValueError: too many values to unpack (expected 2)

```
a, *b= [1, 2,3]
print(a,b)
```

1 [2,3]

```
y[2], y = "abc", "z"
```

```
s = [(i, j) if i < j for i in [1, 2, 3] for j in [1, 2, 3]]
print(s)
```

SyntaxError: expected 'else' after 'if' expression

```
s = "banana"
print(s.replace("na", "xy", 1))
```

```
s = "a,b,c,d"
print(s.split(",", 2))
```

```
s = "a,b,c,d"
print(s.split(",", -1))
```

```
s = "a,b,c,d"
print(s.split(",", 5))
```

Quiz

```
a = b = c = [1, 2]
a.append(3)
print(b, c)
```

[1,2,3] [1,2,3]

```
a, b = [1, 2]
print(a,b)
```

1 2

```
a, b = [1, 2,3]
print(a,b)
```

ValueError: too many values to unpack (expected 2)

```
a, *b= [1, 2,3]
print(a,b)
```

1 [2,3]

```
y[2], y = "abc", "z"
```

NameError: name 'y' is not defined

```
s = [(i, j) if i < j for i in [1, 2, 3] for j in [1, 2, 3]]
print(s)
```

SyntaxError: expected 'else' after 'if' expression

```
s = "banana"
print(s.replace("na", "xy", 1))
```

```
s = "a,b,c,d"
print(s.split(",", 2))
```

```
s = "a,b,c,d"
print(s.split(",", -1))
```

```
s = "a,b,c,d"
print(s.split(",", 5))
```

Quiz

```
a = b = c = [1, 2]
a.append(3)
print(b, c)
```

[1,2,3] [1,2,3]

```
a, b = [1, 2]
print(a,b)
```

1 2

```
a, b = [1, 2,3]
print(a,b)
```

ValueError: too many values to unpack (expected 2)

```
a, *b= [1, 2,3]
print(a,b)
```

1 [2,3]

```
y[2], y = "abc", "z"
```

NameError: name 'y' is not defined

```
s = [(i, j) if i < j for i in [1, 2, 3] for j in [1, 2, 3]]
print(s)
```

SyntaxError: expected 'else' after 'if' expression

```
s = "banana"
print(s.replace("na", "xy", 1))
```

baxyna

```
s = "a,b,c,d"
print(s.split(",", 2))
```

```
s = "a,b,c,d"
print(s.split(",", -1))
```

```
s = "a,b,c,d"
print(s.split(",", 5))
```

Quiz

```
a = b = c = [1, 2]
a.append(3)
print(b, c)
```

[1,2,3] [1,2,3]

```
a, b = [1, 2]
print(a,b)
```

1 2

```
a, b = [1, 2,3]
print(a,b)
```

ValueError: too many values to unpack (expected 2)

```
a, *b= [1, 2,3]
print(a,b)
```

1 [2,3]

```
y[2], y = "abc", "z"
```

NameError: name 'y' is not defined

```
s = [(i, j) if i < j for i in [1, 2, 3] for j in [1, 2, 3]]
print(s)
```

SyntaxError: expected 'else' after 'if' expression

```
s = "banana"
print(s.replace("na", "xy", 1))
```

baxyna

```
s = "a,b,c,d"
print(s.split(",", 2))
```

[‘a’ , ‘b' , ‘c,d']

```
s = "a,b,c,d"
print(s.split(",", -1))
```

```
s = "a,b,c,d"
print(s.split(",", 5))
```

Quiz

```
a = b = c = [1, 2]
a.append(3)
print(b, c)
```

[1,2,3] [1,2,3]

```
a, b = [1, 2]
print(a,b)
```

1 2

```
a, b = [1, 2,3]
print(a,b)
```

ValueError: too many values to unpack (expected 2)

```
a, *b= [1, 2,3]
print(a,b)
```

1 [2,3]

```
y[2], y = "abc", "z"
```

NameError: name 'y' is not defined

```
s = [(i, j) if i < j for i in [1, 2, 3] for j in [1, 2, 3]]
print(s)
```

SyntaxError: expected 'else' after 'if' expression

```
s = "banana"
print(s.replace("na", "xy", 1))
```

baxyna

```
s = "a,b,c,d"
print(s.split(",", 2))
```

['a' , 'b' , 'c,d']

['a' , 'b' , 'c' , 'd']

```
s = "a,b,c,d"
print(s.split(",", -1))
```

```
s = "a,b,c,d"
print(s.split(",", 5))
```

If maxsplit is -1 (or any **negative** number), it means **"no limit"**. The string will be split *completely*.

Formatted String

```python
total_marks = 340
total_sub = 4
print(f"total marks are {total_marks} and number of sub is {total_sub}")
print(f"avg is {total_marks/total_sub:.2f}")
```

```
total marks are 340 and number of sub is 4
avg is 85.00
```

Formatted String

```
1 total_marks = 340
2 total_sub = 4
3 print(f"total marks are {total_marks} and number of sub is {total_sub}")
4 print(f"avg is {total_marks/total_sub:.2f}")
```

No space between f”

Two decimal places (rounded)

```
total marks are 340 and number of sub is 4
avg is 85.00
```

Formatted String

```
1 total_marks = 340
2 total_sub = 4
3 print(f"total marks are {total_marks} and number of sub is {total_sub}")
4 print(f"avg is {total_marks/total_sub:.2f}")
```

No space between f”

Two decimal places (rounded)

```
total marks are 340 and number of sub is 4
avg is 85.00
```

```
1 total_marks = 340
2 total_sub = 4
3 print(f"total marks are {total_marks} and number of sub is {total_sub}")
4 print(f"avg is {total_marks/total_sub:.2f}")
5 print(f"result is {'passed' if total_marks > 300 else 'fail'}")
```

```
total marks are 340 and number of sub is 4
avg is 85.00
result is passed
```

Formatted String

```
1 total_marks = 340
2 total_sub = 4
3 print(f"total marks are {total_marks} and number of sub is {total_sub}")
4 print(f"avg is {total_marks/total_sub:.2f}")
```

No space between f”

Two decimal places (rounded)

```
total marks are 340 and number of sub is 4
avg is 85.00
```

```
1 total_marks = 340
2 total_sub = 4
3 print(f"total marks are {total_marks} and number of sub is {total_sub}")
4 print(f"avg is {total_marks/total_sub:.2f}")
5 print(f"result is {'passed' if total_marks > 300 else 'fail'}")
```

Conditional

```
total marks are 340 and number of sub is 4
avg is 85.00
result is passed
```

Formatted String

```python
total_marks = 340
total_sub = 4
print(f"total marks are {total_marks} and number of sub is {total_sub}")
print(f"avg is {total_marks/total_sub:.2f}")
```

No space between f”

Two decimal places (rounded)

```
total marks are 340 and number of sub is 4
avg is 85.00
```

```python
total_marks = 340
total_sub = 4
print(f"total marks are {total_marks} and number of sub is {total_sub}")
print(f"avg is {total_marks/total_sub:.2f}")
print(f"result is {'passed' if total_marks > 300 else 'fail'}")
```

Conditional

```
total marks are 340 and number of sub is 4
avg is 85.00
result is passed
```

```python
nums = [1, 2, 3]
msg = f"Numbers: {', '.join(str(n) for n in nums)}"
print(msg)
```

Formatted String

```python
total_marks = 340
total_sub = 4
print(f"total marks are {total_marks} and number of sub is {total_sub}")
print(f"avg is {total_marks/total_sub:.2f}")
```

No space between f”

Two decimal places (rounded)

```
total marks are 340 and number of sub is 4
avg is 85.00
```

```python
total_marks = 340
total_sub = 4
print(f"total marks are {total_marks} and number of sub is {total_sub}")
print(f"avg is {total_marks/total_sub:.2f}")
print(f"result is {'passed' if total_marks > 300 else 'fail'}")
```

Conditional

```
total marks are 340 and number of sub is 4
avg is 85.00
result is passed
```

```python
nums = [1, 2, 3]
msg = f"Numbers: {', '.join(str(n) for n in nums)}"
print(msg)
```

```
Numbers: 1, 2, 3
```

Parse Input

Keep track of marks until empty line is entered. Calculated sum and avg.

```
1  line = input("name and marks:")
2  sum = 0
3  count = 0
4  while line != "":
5      marks = line.split()[1]
6      sum += int(marks) if marks.isdecimal() else 0
7      count += 1
8      line = input("name and marks:")
9  print(f"sum is {sum}")
10 print(f"avg is {sum/count:.2f}")
11
```

```
name and marks:priyanka 40
name and marks:subodh 50
name and marks:svs 60
name and marks:sayan 70
name and marks:
sum is 220
avg is 55.00
```

Parse Input

Keep track of marks until empty line is entered. Calculated sum and avg.

```
1 line = input("enter name and marks")
2 sum = 0
3 count = 0
4 while line!= "":
5     marks = line.split()[1]
6     if not str(marks).isdecimal():
7         line = input("enter name and marks")
8         continue
9     marks =  int(marks)
10    sum += marks
11    count += 1
12    line = input("enter name and marks")
13 print(f"total is {sum}")
14 print(f"total is {sum/count}")
```

```
enter name and markspriyanka 20
enter name and markssayan 30
enter name and markssubodh 40
enter name and markssvs 50
enter name and markspriyanka hello
enter name and marks
total is 140
total is 35.0
```