1st Midterm Exam

2023.04.13 (Thursday) 10:10 – 12:00

1. **[Python Basics]** In each of the following questions, you are asked to show what will be printed out. If there is an error, please explain why it is an error. (12%)

(a)	`print(9/5, "vs.", 9//5, "vs.", 9%5)`
(b)	`print(4 * 12 / 2 ** 2 - 2)`

(c)	`m, x, b = 2, 3, 5`
	`y = mx + b`
	`print(y)`

(d)	`alpha, x = 0.5, 10`
	`cofactor = alpha * x * x`
	`1storder = 1.0/cofactor`
	`print(1storder)`

(e)	`a, b = 3, 5`
	`a, b = b, a`
	`b, a = a, b`
	`print(a, b)`

(f)	`s = "3.14"` `print(int(float(s)))`
(g)	`word = 'Code!'` `print(word[-1] + word[5])`

(h)	`s = 'cat'`
	`s[0] = 'h'`
	`print(s)`

(i)	`print("Time left %02d:%05.2f" % (9, 30.19283))`
(j)	`s1, s2 = "ok", "go"` `print(3*s1 + s2 + 3)`
(k)	`s = "okokgo"` `print(s[2:5])`
(l)	`s = "okokgo"` `print(s[::-1])`

2. **[Conditionals & Loops]** In each of the following questions, you are asked to show what will be printed out. If there is an error, please explain why it is an error. (16%)

(a)
```
happy, money = True, 0
print(not happy or not money)
```

(b)
```
print(('3' + 3) == 6 and True)   # True or False
```

(c)
```
print(1 + (4 and 6) + (5 or 0) + (0 and 8))
```

(d)
```
score = 20
while score > 1:
    score = score/2 - 1
print(score)
```

(e)
```
x = 5
y = 5
while x <= y and y <= x:
    x = x + 1
    y = y + 1
print("Done!")
```

(f)
```
n = 5
a, b, i = 1, 1, 1
while i < 5:
    a, b = b, a + b
    i += 1
print(a)
```

(g)
```
a, b = 84, 48
while b:
    a, b = b, a % b
print(a)
```

(h)
```
n = 12
while n != 1:
    print(n, end=", ")
    if n % 2 == 0:
        n = n // 2
    else:
        n = n * 3 + 1
print(n, end=".\n")
```

3. **[List]** In each of the following questions, you are asked to show what will be printed out. If there is an error, please explain why it is an error. (10%)

(a)
```
d1 = ["Kobe", "Bryant", 23]
d2 = "Kobe Bryant 23"
d1[2] = "24"
d2[12:] = "24"
print(d1, d2)
```

(b)
```
scores = [80, 70, 60, 90]
print((sum(scores)/len(scores)))
```

(c)
```
game = ["rock", "paper"]
game = game + "scissor"
print(game)
```

(d)
```
n = [1, 3, 5, 2, 4, 6]
print(n[1:4] * 2)
```

(e)
```
notes = ["Do", "Re", "Mi"]
print(notes[2] == "Mi")  # True or False
```

(f)
```
list = [1, 2, 3]
list.append(4)
list.append([5])
list.extend([6])
print(list)
```

(g)
```
data = "I love Python"
print(data.split(" "))
```

(h)
```
v = [1, 2, 3, 4]
p = list(v)
p[2] = 5
print(p)
print(v)
```

(i)
```
v = [1, 2, 3, 4]
p = v
p[2] = 5
print(p)
print(v)
```

(j)
```
x = [5, 3, 2, 6, 1]
x.pop()
x.remove(6)
x.insert(0, 0)
x.sort(reverse = True)
print(x)
```

4. Please read the following code (left block), and answer what will be printed out based on the respective initialized x in questions (a) – (d). (4%)

```
# initialize x here
if x % 2 != 0:
    if x ** 2 <= 36:
        print('pow')
    else:
        print(x // 3)
else:
    if x < 0:
        print(False)
    elif not x + 2 > 8:
        print(x / 2)
    else:
        print("end")
```

Questions	The initialized x	Printed?
(a)	`x = 2`	
(b)	`x = 13`	
(c)	`x = -8`	
(d)	`x = 10`	

5. Fill in two blanks the following code. This program is supposed to allow the user to input a string, remove the consecutive repeated characters, and print the result. The sample input and output is shown in the table. If you don't know how to complete these two blanks, you can write your own code to generate the printed output based on the corresponding input and output in the following examples. (8%)

Example 1	**Example 2**
`Input: hhhellloooooo` `Output: helo`	`Input: 55555----444433322...1` `Output: 5-432.1`

```
s = input("Input: ")
result = ''     # it is used to store the result
i = 0
while [                    ]:
    current = s[i]
    result = result + current
    i = i + 1
    while [                                ]:
        i = i + 1
print("Output", result)
```

6. Write a program that allows the user to input a time string in the format `'h:m:s'`, in which h is hour $(0 \leq h \leq 23)$, m is minute $(0 \leq m \leq 59)$, and s is second $(0 \leq s \leq 59)$. Then your program should convert time `'h:m:s'` to seconds in a day, and print the converted seconds. The sample input and output are shown in the following. (10%)

Example 1	`Enter a time (h:m:s): 1:10:25` `Seconds in a day = 4225`
Example 2	`Enter a time (h:m:s): 15:3:8` `Seconds in a day = 54188`

7. Write a program that allows the user to input a positive integer n, then find and print the **list** of prime numbers from 2 to n. (10%)

Example 1	`Enter a positive integer: 10` `Prime list (2-10): [2, 3, 5, 7]`
Example 2	`Enter a positive integer: 20` `Prime list (2-20): [2, 3, 5, 7, 11, 13, 17, 19]`

8. Please write a program with nested `while` loops (e.g., `while` in `while` in `while`) to generate and print a 9×9 multiplication table in a particular order, **as exactly shown in the right figure**. Note that if you totally use `print()` (i.e., without using while loops) to generate the multiplication table, you will get only 1%. (10%)

```
9 x 1 = 9    6 x 1 = 6    3 x 1 = 3
9 x 2 = 18   6 x 2 = 12   3 x 2 = 6
9 x 3 = 27   6 x 3 = 18   3 x 3 = 9
9 x 4 = 36   6 x 4 = 24   3 x 4 = 12
9 x 5 = 45   6 x 5 = 30   3 x 5 = 15
9 x 6 = 54   6 x 6 = 36   3 x 6 = 18
9 x 7 = 63   6 x 7 = 42   3 x 7 = 21
9 x 8 = 72   6 x 8 = 48   3 x 8 = 24
9 x 9 = 81   6 x 9 = 54   3 x 9 = 27

8 x 1 = 8    5 x 1 = 5    2 x 1 = 2
8 x 2 = 16   5 x 2 = 10   2 x 2 = 4
8 x 3 = 24   5 x 3 = 15   2 x 3 = 6
8 x 4 = 32   5 x 4 = 20   2 x 4 = 8
8 x 5 = 40   5 x 5 = 25   2 x 5 = 10
8 x 6 = 48   5 x 6 = 30   2 x 6 = 12
8 x 7 = 56   5 x 7 = 35   2 x 7 = 14
8 x 8 = 64   5 x 8 = 40   2 x 8 = 16
8 x 9 = 72   5 x 9 = 45   2 x 9 = 18

7 x 1 = 7    4 x 1 = 4    1 x 1 = 1
7 x 2 = 14   4 x 2 = 8    1 x 2 = 2
7 x 3 = 21   4 x 3 = 12   1 x 3 = 3
7 x 4 = 28   4 x 4 = 16   1 x 4 = 4
7 x 5 = 35   4 x 5 = 20   1 x 5 = 5
7 x 6 = 42   4 x 6 = 24   1 x 6 = 6
7 x 7 = 49   4 x 7 = 28   1 x 7 = 7
7 x 8 = 56   4 x 8 = 32   1 x 8 = 8
7 x 9 = 63   4 x 9 = 36   1 x 9 = 9
```