

**QUEENSBOROUGH COMMUNITY COLLEGE**  
**The City University of New York**  
**Department of Engineering Technology**

---

**Programming Exercises – Functions I**

1. `input()` & `fmod()`

- a) Use `input()` function to request any two numbers.
- b) Use `math` module, `fmod()` to return the remainder of the user input.
- c) Print out the result as an integer.
- \*d) Implement the validation of denominator to be a non-zero number in a sentinel-controlled loop.

Example Output 1

```
Enter a numerator: 10
Enter a denominator: 2
10 mod 2 = 0
```

Example Output 2

```
Enter a numerator: 1
Enter a denominator: 0
Denominator cannot be zero. Try again.
```

```
Enter a numerator: 0
Enter a denominator: 1
0 mod 1 = 0
```

2. `randint()` & `isqrt()`

- a) Use `random` module, `randint()` to generate a random number in the range (1, 100).
- b) Use `math` module, `isqrt()` to round a square root number downwards to the nearest integer.
- c) Print out the result.

Example Output 1

```
Square root of 4 = 2
```

Example Output 2

```
Square root of 8 = 2
```

3. Write a function `hello()` that prints “Hello World” to the console. Implement the code to test the function.

4. Modify the function, `hello()` above with a parameter.

- a) Define the function, `helloNo(n)` with a loop to call `hello()`  $n$  times to the console.
- b) Use the parameter,  $n$  for the numbers of iterations in the loop.

Example Output: `helloNo(3)` will print the following

```
Hello World
Hello World
Hello World
```

5. Write a program that creates a **void** function to display a given message. Implement the code to test the function.

a) *message(p1, p2)* uses a loop to print the text stored in p1, p2 times to the console.

b) Define a *main()* function to do the following:

- 1) Request and print an input *text* from the console and print the *text*.
- 2) Get a random integer number, *n*, in the range (1, 10) and print *n*.
- 3) Call *message()* function with arguments, *text* and *n*.
- 4) Handle all input and output.

c) Call *main()* function to initiate the tasks to be performed.

Example Output

```
Enter a text: looking ahead to spring
text = Looking Ahead To Spring
n = 3
message(text, n) will print the following:
Looking Ahead To Spring
Looking Ahead To Spring
Looking Ahead To Spring
```

6. Write a program that creates a **list-returned** function to display a list contains all but the first and last elements.

Implement the code to test the function.

a) Define a function, *middle(l)* with a list as the parameter:

- 1) *middle(l)* slices and constructs the list parameter.
- 2) *middle(l)* **returns** a new list that contains all but the first and last elements.  
*For example, middle ([1,2,3,4]) should return [2,3].*

b) Define a *main()* function to do the following:

- 1) Create a list, *numList* with *n* numbers in the list.
- 2) Get a random integer, *n*, in the range (1, 10).
- 3) Call *middle(numList)* function and print the returned list.
- 4) Handle all input and output.

c) Call *main()* function to initiate the tasks to be performed.

Example Output 1

```
List length = 4
[1, 2, 3, 4]
[2, 3]
```

Example Output 2

```
List length = 2
[1, 2]
[]
```

Example Output 3

```
No change made to the list.
List length = 1
[1]
[1]
```