

# VG101 Lab Worksheet

## Lab 2

Instructor: Dr. Yifei ZHU

TA: Hangrui CAO

TA: Qinhang WU

TA: Muchen XU

## Code Quality Requirement

- Ensure proper indentation
- Make naming meaningful
- Include necessary comments
- Split the code over functions
- Test the code as much as you can

Category	Exercise title	Label
basic syntax exercise	Odd or Even 1.0	condition
basic syntax exercise	GPA Converter	condition
basic syntax exercise	Simple Sum	loop
basic syntax exercise	Unknown Length	loop
basic syntax exercise	Odd or Even 2.0	loop
basic syntax exercise	Repeated Number*	nested loop
basic syntax exercise	Prime or Not*	loop control
basic syntax exercise	Construction or Enumeration*	loop control
practical exercise	Arbitrary-Precision Plus*	string, loop
practical exercise	Daffodil Number*	loop, enumeration
practical exercise	Base Conversion	loop, base
practical exercise	Sort*	loop, sort
challenge	Distinct Powers	loop, number theory

- In-lab quiz will test your understanding of some of these problems, get prepared.
- Challenge is totally optional, and you will be rewarded a bonus if you do a correct implementation and submit the result on time.
- Remember to submit your code on time, otherwise your attendance score will be deducted.
- Coding is not that difficult. Think before programming to see whether there is an easier implementation. Make your life easier.
- Problems labeled with \* are recommended.

## Basic Syntax Exercise

### Conditional Expression

#### Odd or Even 1.0

Read an integer from command line, then display 1 if it's even, else display 0.

Example: (You can modify the file name or the prompt as you like)

```
1  >> ex1
2  Input an integer: 3
3      0
4
5  >> ex1
6  Input an integer: 2
7      1
```

## GPA Converter

Read an integer as the score from command line, then display the letter grade according to the SJTU GPA table.

Example:

```
1  >>ex2
2  95
3      A+
4
5  >>ex2
6  78
7      B
```

## SJTU GPA Table

Score	Letter Grade	Grade points
95~100	A+	4.3
90~94	A	4.0
85~89	A-	3.7
80~84	B+	3.3
75~79	B	3.0
70~74	B-	2.7
67~69	C+	2.3
65~66	C	2.0
62~64	C-	1.7
60~61	D	1.0
<60	F	0

## Loop Expression

### Simple Sum

Read an integer as  $n$ , and then calculate,

$$\sum_{i=1}^n i^3$$

Example:

```
1 >>ex3
2 Input an integer: 3
3      36
```

## Unknown Length

Read a sequence of integers ending with 0, find the length of the sequence. Note that this example provides you with an idea of reading in data of unknown length.

Example:

```
1 >>ex4
2 1
3 2
4 3
5 4
6 0
7      5
```

## Odd or Even 2.0

Read a sequence of integers ending with 0, check whether every integer is odd or even.

Example:

```
1 >>ex5
2 1
3      1
4
5 2
6      0
7
8 3
9      1
10
11 4
12      0
13
14 0
```

## Repeated Number

Read a sequence of integers ending with 0, find whether there are repeated number, i.e. an integer occurred twice or more.

Example:

```
1 >>ex6
```

```

2 | 1
3 | 3
4 | 4
5 | 5
6 | 2
7 | 0
8 |      0
9 |
10 | >>ex6
11 | 1
12 | 3
13 | 4
14 | 5
15 | 3
16 | 3
17 | 4
18 | 0
19 |      1

```

## Break and Continue

### Prime or Not

Read an positive integer, determine whether it is a prime number.

You may use a variable `flag` to store your result.

```

1 | >>ex7
2 | 233
3 |      1
4 |
5 | >>ex7
6 | 232
7 |      0

```

### Construction or Enumeration

Read a sequence of positive integers ending with 0. For each number  $n$ , find an arbitrary  $m$  such that  $nm + 1$  is not a prime number and  $m > 1$ .

```

1 | >>ex8
2 | 10
3 |      2
4 |
5 | 6
6 |      4
7 |
8 | 20
9 |      4

```

## Practical Exercises

### Arbitrary-precision Plus

Input two large integers (firstly store as strings). Display the sum of two integers.

```
1 >>ex9
2 123456789123456789123456789
3 987654321987654321987654321
4 111111111111111111111111111110
```

Consider:

- How do you perform arithmetic plus by "vertical form"?
- Should we compute from highest digits or lowest digits?
- How to handle the "carry"?
- What if their length are different?

## Daffodil Number

---

The daffodil number is one of the famous interesting numbers in the mathematical world. A daffodil number is a three-digit number whose value is equal to the sum of cubes of each digit.

For example. 153 is a daffodil as  $153 = 1^3 + 5^3 + 3^3$ .

Find all the three-digit daffodil number (larger than 100).

Consider:

- First write a piece code to check whether a three-digit number is "daffodil".
- Use that code to check every possible three-digit number.

## Base Conversion

---

Input an integer, convert it to binary base. Display the answer as a string.

```
1 >>ex11
2 31
3 11111
```

## Sort

---

Read a vector consists of several integers, output them in the ascending order.

Example:

```
1 >>ex13
2 [2 3 1 4]
3 1 2 3 4
```

Consider:

- Find the minimum of (the rest of) the sequence, and then place it to somewhere else.

## Advanced Exercise - Distinct Powers

---

### Description

---

Consider all integer combinations of  $a^b$  for  $2 \leq a \leq 5$  and  $2 \leq b \leq 5$ :

$$\begin{aligned}2^2 &= 4, 2^3 = 8, 2^4 = 16, 2^5 = 32 \\3^2 &= 9, 3^3 = 27, 3^4 = 81, 3^5 = 243 \\4^2 &= 16, 4^3 = 64, 4^4 = 256, 4^5 = 1024 \\5^2 &= 25, 5^3 = 125, 5^4 = 625, 5^5 = 3125\end{aligned}$$

If they are lined up, with any repeats removed, we get the following sequence of 15 distinct terms:

4, 8, 9, 16, 25, 27, 32, 64, 81, 125, 243, 256, 625, 1024, 3125

## Problem Statement

---

How many distinct terms are in the sequence generated by  $a^b$  for

- $2 \leq a \leq 10$  and  $2 \leq b \leq 10$ ? (0.05 marks)
- $2 \leq a \leq 100$  and  $2 \leq b \leq 100$ ? (0.05 marks)
- $2 \leq a \leq 100000$  and  $2 \leq b \leq 100000$ ? (0.05 marks)

**Improve the efficiency as possible as you can!**