

GET READY  
FRESHY

2110101

COM PROG

By Worralop Srichainont

# Introduction

GET READY  
**FRESHY**



**Bloom #2 CP**

**Worralop Srichainont**



bloom.wrls



Worralop Srichainont



reisenx3395



# Course Syllabus

## รายละเอียดรายวิชา

อ้างอิงตาม Course Syllabus ของปีการศึกษา 2566 ภาคการเรียนที่ 2

# วิชานี้เรียนอะไร



## **2110101 COMPUTER PROGRAMMING**

- เรียนเกี่ยวกับการเขียนโปรแกรมภาษา Python พื้นฐาน
- เน้นการเขียนโปรแกรมเพื่อแก้ไขโจทย์ปัญหาคอมพิวเตอร์
- วิชานี้เก็บคะแนนจากการสอบ Grader เป็นหลัก



# การเก็บคะแนน

รายการ	คะแนน
สอบเก็บคะแนน Grader	45
สอบกลางภาค	15
สอบปลายภาค	20
การบ้านเขียนโปรแกรม	10
งานในห้องเรียน	10
<b>รวม</b>	<b>100</b>



- การสอบ Grader มี 3 ครั้ง ครั้งละ 15 คะแนน
- การบ้านเขียนโปรแกรมมี ทั้งหมด 5 ครั้ง
- งานในห้องเรียน จะแตกต่างตาม Section ที่เรียน
- ภาครคอม (CP) ไม่มีการบ้านเขียนโปรแกรม แต่ว่าเป็น Workshop แทน

# การตัดเกรด

GET READY  
**FRESHY**  
กบก.

คะแนนรวม	เกรด
80.00 ขึ้นไป	A
75.00 - 79.99	B+
70.00 - 74.99	B
65.00 - 69.99	C+

คะแนนรวม	เกรด
60.00 - 64.99	C
55.00 - 59.99	D+
50.00 - 54.99	D
ต่ำกว่า 50.00	F

# ເບື້ອຫາທີ່ເຮັດ

GET READY  
FRESHY



unit 01: Data Type & Expression

unit 02: Basic String & List

unit 03: Selection

unit 04: Repetition

unit 05: List Processing

unit 06: Function

unit 07: String Processing

unit 08: Basic Dict

unit 09: Nested Structure

unit 10: Tuple, Set, Dict

unit 11: NumPy

unit 12: Class & Object

GET READY  
**FRESHY**



# Grader Exam

สอบเก็บคะแนนเกรดเดอร์

# Grader ค้อวะໄສ



**Grader** เป็นระบบตรวจสอบว่า code ที่ส่งไปถูกต้องตามแนวคิดของโจทย์หรือไม่

The screenshot shows the Grader application interface. At the top, there's a blue header bar with a house icon and the word "MAIN". On the right side of the header is a menu icon. Below the header, there's a sidebar on the left labeled "โจทย์" (Assignment) with two items:

- ไฟล์โจทย์
- ระดับความยาก

Two assignment cards are displayed:

- Assignment 35: 05\_Nicknames ★**
  - Details: 5 days ago (06/03/24 20:40:29), 100.0 [PPPPPPPPPP], compiler msg, src, submissions (2), testcases.
  - Status: Score 100.00 (green progress bar).
  - Actions: Edit button, Read link.
- Assignment 36: 05\_Back\_n\_Front ★**
  - Details: 2 days ago (09/03/24 23:22:31), 40.0 [PxxPPxxxxP], compiler msg, src, submissions (2), testcases.
  - Status: Score 40.00 (yellow progress bar).
  - Actions: Edit button, Read link.

## ส่ง Code

- แก้ไข Code ที่ส่ง
- คำແບບ

# Grader คืออะไร

GET READY  
FRESHY

กบค.

## ตัวอย่างโจทย์ใน Grader

### โครงเป็นพี่

จงเขียนโปรแกรมรับ ชื่อเล่น เดือน วัน ปี เกิด ของคนสองคน และแสดงว่าใครอายุมากกว่า

### ข้อมูลนำเข้า

ข้อมูลชื่อและวันเดือนปีเกิดของสองคน คนละบรรทัด ในรูปแบบ

ชื่อเล่น ชื่อเดือน เลขวัน, เลขปี พ.ศ.

### ข้อมูลส่งออก

ชื่อเล่นของผู้ที่เกิดก่อน ในกรณีที่อายุเท่ากัน ให้แสดงทั้งสองชื่อ เรียงตามลำดับที่อ่านเข้ามา

input (จากแป้นพิมพ์)	output (ทางจอภาพ)
Jane March 23, 2543	Jane
Kate June 9, 2544	
Jib July 4, 2545	Jib
Mobile July 9, 2545	

# Grader คืออะไร

**Grader** จะให้คะแนนตามกรณีทดสอบ (testcases) ที่ถูกต้อง

MAIN Submissions ▾

Live submit

```
1 t = input()
2 if t != "str2RLE" and t != "RLE2str":
3     print("Error")
4
5 w = input()
6 w += " " # จะได้ทำตัวห่างสุด
7 c = 0
8 f = w[0]
9 s = ""
10 z = []
11
12 if t == "str2RLE":
13     for e in w:
14         if f == e:
15             c += 1
16
17     else:
18         s += str(f) + " " + str(c) + " "
19         f = e
20         c = 0
21         c += 1
22
23     print(s)
24 elif t == "RLE2str":
25     x = w.split()
26     for i in range(0, len(x), 2):
27         z.append(str(x[i]) * int(x[i+1]))
```

ลอง

Submission

Task: Part-I-RLE  
P1\_02\_RLE  
1.0 (s) | 512 (MB)

Description: [Read]

Language: Python

เลือกไฟล์: ไม่ได้เลือกไฟล์ใด

Submit

Latest Submission Status Refresh

Sub ID: 1893534  
3 minutes ago (22:46:49)  
92.3077 [PPPPPPPPPPPx]

compiler msg src submissions (8) testcases

ส่วนแก้ไข Code

คะแนนของ Code



กบค.

## การอ่านคะแนน Grader

Sub ID	1944108
🕒	less than a minute ago (04:29:52)
⌚	8.3333 [-xxP-x-T-xx-]

### ตัวอักษร 1 ตัว แทน 1 Testcase โดยที่

- **P** คือ Code ของเรามี Output ถูกต้องตาม Testcase
- **-** คือ Code ของเรามี Output ไม่ถูกต้องตาม Testcase
- **x** คือ มีการใช้งานคำสั่งที่ไม่ถูกต้อง
- **T** คือ Code ของเราใช้เวลา Run นานเกินไป

# สอบ Grader



## การสอบ Grader มีการสอบเก็บคะแนนทั้งหมด 3 ครั้ง

- Grader 01 เก็บ 15 คะแนน
- Grader 02 เก็บ 15 คะแนน
- Grader 03 เก็บ 15 คะแนน

**รวมทั้งหมด 45 คะแนน**

**สอบที่ศูนย์คอมพิวเตอร์ คณะวิศวกรรมศาสตร์**



# สอบ Grader

GET READY  
FRESHY



## ลักษณะการสอบ Grader แต่ละครั้ง

- การสอบ Grader แต่ละครั้งมีระยะเวลาทั้งหมด 90 นาที
- การสอบแต่ละครั้งจะมีข้อสอบทั้งหมด 3 ข้อ ข้อละ 100 คะแนน
- ข้อสอบจะถูกส่งมาพร้อมกันทั้ง 3 ข้อ
- สามารถเริ่มทำข้อไหนก่อนก็ได้ ส่ง code กี่รอบก็ได้ ไม่จำกัดจำนวนครั้ง
- นำกระดาษเข้าห้องสอบได้ 1 แผ่น จะจดอะไรเข้าไปก็ได้
- สอบที่ศูนย์คอมพิวเตอร์ คณะวิศวกรรมศาสตร์

**\*\*ใช้ Windows XP และ Python version 3.4.4 ในการสอบ**

# สอบ Grader

GET READY  
FRESHY



## การคำนวณคะแนน Grader

ข้อสอบมี 3 ข้อ ข้อละ 100 คะแนน แต่จะเก็บคะแนนจริง 400 คะแนน โดยจะเอาข้อที่ได้คะแนนสูงที่สุดมาคูณ 2 และข้ออื่นคิดคะแนนตามเดิม

### ตัวอย่างการคำนวณคะแนน

Grader	คะแนน
Grader 1-1	40
Grader 1-2	<b>80(x2)</b>
Grader 1-3	10
<b>รวม</b>	<b>210</b>

Grader	คะแนน
Grader 1-1	<b>100(x2)</b>
Grader 1-2	100
Grader 1-3	100
<b>รวม</b>	<b>400</b>

# Grader 01



## เนื้อหาที่ออกในการสอบ Grader 01 (15 คะแนน)

- บทที่ 01: Data Type & Expression
- บทที่ 02: Basic String & List
- บทที่ 03: Selection
- บทที่ 04: Repetition

**\*\*Grader 01 จะออก Flowchart แบบน่อน 1 ข้อ**

# Grader 01

GET READY  
FRESHY

กวดค.  
 $\Sigma$

## ข้อสอบ Grader 01 ปี 2566 เทอม 2

### Repeat Count

จงเขียนโปรแกรมนับจำนวน pattern ที่อยู่เรียงติดกัน 2 ครั้งขึ้นไปในข้อมูลที่รับเข้ามา โดย pattern จะมีคุณสมบัติดังนี้

- เป็นอักขระ A ถึง Z ตัวพิมพ์ใหญ่เท่านั้น (รับประกันว่าไม่มีอักขระอื่น) ความยาว 1-5 ตัวอักษร
- อาจจะมีอักขระที่ซ้ำกันได้ ตัวอย่างเช่น GRADE, DOG, BBB, D

โดยข้อมูลที่รับเข้ามาจะมีคุณสมบัติดังนี้

- เป็นอักขระ A ถึง Z ตัวพิมพ์ใหญ่เท่านั้น (รับประกันว่าไม่มีอักขระอื่น) ความยาวอย่างน้อย 1 ตัวอักษร

เพื่อไม่ให้เป็นการยากจนเกินไปการเรียงติดกันที่เป็นคำตอบ จะมีอยู่ไม่เกิน 1 ที่

- จะไม่มีกรณีที่ข้อมูลที่รับเข้ามาเป็น "TGAGAGATTGAGAGACT" และ pattern คือ GA

### ข้อมูลนำเข้า

บรรทัดแรกเป็น pattern

บรรทัดถัดไปเป็นจำนวนเต็ม N แทนข้อมูลที่ต้องการตรวจสอบ

อีก N บรรทัดถัดมา แต่ละบรรทัดจะเป็นข้อมูลที่ต้องการให้นับจำนวน pattern ที่อยู่เรียงติดกัน 2 ครั้งขึ้นไป

### ข้อมูลส่งออก

N บรรทัด แต่ละบรรทัดแสดงจำนวนการติดกันของ pattern

# Grader 01



## ข้อสอบ Grader 01 ปี 2566 เทอม 2

ตัวอย่าง

input (จากแป้นพิมพ์)	output (ทางจอกาพ)
ABC	2
5	3
ABCABC	0
ABCABCABC	0
ABC	3
ABCDABCDABCD	
ABCABCABCDABCDABCD	
B	0
5	2
B	3
BB	0
BBB	4
ABABABABA	
ABBBBABABAB	

# Grader 01

GET READY  
FRESHY



## ເລຍຂ້ວສອບ Grader 01 ປີ 2566 ແລ້ມ 2

```
1  pattern = str(input())
2  n = int(input())
3
4  repeat_list = []
5  for i in range(n):
6      text = str(input())
7      repeat = len(text)//len(pattern)
8      while(repeat >= 2):
9          if(pattern*repeat in text):
10              repeat_list.append(repeat)
11              break
12          else:
13              repeat -= 1
14      if(repeat < 2):
15          repeat_list.append(0)
16
17  for item in repeat_list:
18      print(item)
```

# Grader 02



## **เนื้อหาที่ออกในการสอบ Grader 02 (15 คะแนน)**

- บทที่ 05: List Processing
- บทที่ 06: Function
- บทที่ 07: String Processing
- บทที่ 08: Basic Dict

# Grader 02



## ข้อสอบ Grader 02 ปี 2566 เทอม 2

### ข้อภัยที่มีคำไม่สุภาพในโจทย์นี้

การแสดงข้อความที่มีคำไม่สุภาพอาจไม่เป็นที่ยอมรับในบางสื่อ จึงมักหลีกเลี่ยงการแสดงคำที่ไม่สุภาพด้วยการแทนสาระในคำนั้นด้วยเครื่องหมายดอกจัน เช่น **Biden was caught on a live mic saying "What a st\*p\*d son of a b\*tch."**

ให้เขียนโปรแกรมอ่านรายการคำไม่สุภาพต่าง ๆ ตามด้วยข้อความอีกหนึ่งบรรทัด เพื่อแสดงข้อความใหม่ที่ได้จากการแทนสาระในสตริงย่อของข้อความที่เป็นคำไม่สุภาพด้วยเครื่องหมายดอกจัน

### ข้อมูลนำเข้า

บรรทัดแรกรายการของคำไม่สุภาพต่าง ๆ คั่นด้วยช่องว่าง  
บรรทัดที่สองเป็นข้อความหนึ่งบรรทัด

### ข้อมูลส่งออก

เหมือนข้อความที่รับมาในบรรทัดสอง แต่แทนสาระในสตริงย่อของข้อความที่เป็นคำไม่สุภาพ (ในบรรทัดแรก) ด้วยเครื่องหมายดอกจัน

# Grader 02

GET READY  
FRESHY



## ข้อสอบ Grader 02 ปี 2566 เทอม 2

ตัวอย่าง

input (จากแป้นพิมพ์)	output (ทางจอภาพ)
Crap  He's so full of crap.	He's so full of cr*p.
damn  Damnit, I don't give a damn.	D*mnit, I don't give a d*mn.
crap DAMN  !!!CrapDAMNcRAPdamn_crab_darn!!!	!!!Cr*pD*MNcR*Pd*mn_crab_darn!!!

# Grader 02



## ເລຍຂ້ອສອບ Grader 02 ປີ 2566 ເອມ 2

```
1 def hide_vowel(w):
2     h = ""
3     for c in w:
4         if c.lower() in 'aeiou':
5             c = '*'
6             h += c
7     return h
```

```
24 Owords = input().split()
25 text = input()
26 censor = text
27 for word in Owords:
28     censor = less_offensive(censor,word)
29 print(censor)
```

```
9 def less_offensive(t,oword):
10    start = 0
11    end = len(oword)
12    censor = ""
13    while(start < len(t)):
14        if(t[start:end].lower() == oword.lower()):
15            censor += hide_vowel(t[start:end])
16            start += len(oword)
17            end += len(oword)
18        else:
19            censor += t[start]
20            start += 1
21            end += 1
22    return censor
```

# Grader 03



## เนื้อหาที่ออกในการสอบ Grader 03 (15 คะแนน)

- หน้าที่ 09: Nested Structure
- หน้าที่ 10: Tuple, Set, Dict

# Grader 03

GET READY  
FRESHY

กบค.

## ข้อสอบ Grader 03 ปี 2566 เทอม 2

### Peak 2D

ข้อมูล 2 มิติขนาด  $R \times C$  สามารถเขียนแทนด้วยลิสต์ขนาด  $R$  ช่อง โดยที่แต่ละช่องเก็บลิสต์ของจำนวนเต็ม  $C$  ตัว จงเขียนฟังก์ชัน `count_peak(data)` ที่คืนค่าเป็นจำนวนเต็มแทนจำนวนจุดสูงสุดจากข้อมูล 2 มิติที่กำหนดให้ โดย “จุดสูงสุด” คือ ข้อมูลที่มีค่ามากกว่าข้อมูล ด้านซ้าย ด้านขวา ด้านบน และด้านล่าง (แปลว่าถ้าข้อมูลอยู่ริมขอบด้านใดด้านหนึ่ง หรือมุมด้านใดด้านหนึ่ง จะไม่นับเป็นจุดสูงสุด)

หาก `data` มีค่าเป็น `[[0, 1, 2, 3], [1, 9, 2, 3], [2, 1, 8, 3], [3, 3, 3, 3]]` จะสามารถแสดงเป็น 2 มิติได้เป็น

0	1	2	3
1	9	2	3
2	1	8	3
3	3	3	3

จุดริมขอบ จะไม่นับเป็นจุดสูงสุด

8 นับเป็นจุดสูงสุด เพราะว่ามีค่ามากกว่า ข้อมูลด้านบน (2) ด้านซ้าย (1)  
ด้านล่าง (3) ด้านขวา (3)

จุดสูงสุดมี 2 จุดได้แก่ 9 8

# Grader 03



## ข้อสอบ Grader 03 ปี 2566 เทอม 2

ตัวอย่าง

input (จากแป้นพิมพ์)	output (ทางจอภาพ)
<pre>dat=read_data();print(count_peak(dat)) 4 0 1 2 3 1 9 2 3 2 1 8 3 3 3 3 3</pre>	2
<pre>dat=read_data();print(count_peak(dat)) 3 0 0 0 0 1 0 0 0 0</pre>	1

# Grader 03

GET READY  
FRESHY



## ເລຍຂ້ອສອບ Grader 03 ປີ 2566 ເອມ 2

```
1  def read_data():
2      dat = []
3      R = int(input())
4      for r in range(R):
5          dat.append([int(e) for e in input().strip().split()])
6      return dat
7
8  def count_peak(data):
9      count = 0
10     nrow = len(data)
11     ncol = len(data[0])
12     for i in range(1,nrow-1):
13         for j in range(1,ncol-1):
14             if(data[i][j] > data[i-1][j] and
15                 data[i][j] > data[i+1][j] and
16                 data[i][j] > data[i][j-1] and
17                 data[i][j] > data[i][j+1]):
18                 count += 1
19
20     return count
21 exec(input().strip())
```

GET READY  
**FRESHY**



# Midterm - Final

## สอบมิดเทอมและไฟนอล

# Midterm



## ลักษณะข้อสอบ Midterm

- การสอบ Midterm จะเก็บ 15 คะแนน
- เป็นการสอบปรนัย 5 ช้อยส์ จำนวน 60 ข้อ
- ระยะเวลาการสอบ 2 ชั่วโมง

**\*\*ข้อสอบ Midterm ไม่ยาก แต่ต้องรอบคอบ อ่านโจทย์ดีๆ**

# Midterm



## **เนื้อหาที่ออกในข้อสอบ Midterm**

บทที่ 01: Data Type & Expression

บทที่ 02: Basic String & List

บทที่ 03: Selection

บทที่ 04: Repetition

บทที่ 05: List Processing

บทที่ 06: Function

# Midterm

GET READY  
FRESHY

Σ  
กบค.

## ตัวอย่างข้อสอบ Midterm

ใช้โปรแกรมข้างล่างนี้ ตอบคำถาม 3 ข้อต่อไปนี้

```
t = input().split();
a,b,c = int(t[0]), int(t[1]), int(t[2])
if a > b :
    a += b
elif a < b :
    b += a
else:
    a = b + c
print(a,b,c)
```

14. ถ้าโปรแกรมข้างบนทำงาน แล้วป้อนข้อมูลเป็น **2 4 2** จะแสดงอะไรทางจอภาพ

- A) 2 4 2      B) 2 6 2      C) 6 2 2      D) 6 4 2      E) ไม่มีข้อใดถูก

15. ถ้าโปรแกรมข้างบนทำงาน แล้วป้อนข้อมูลเป็น **4 4 2** จะแสดงอะไรทางจอภาพ

- A) 2 4 2      B) 2 6 2      C) 6 2 2      D) 6 4 2      E) ไม่มีข้อใดถูก

16. ถ้าโปรแกรมข้างบนทำงาน แล้วป้อนข้อมูลเป็น **4 2 2** จะแสดงอะไรทางจอภาพ

- A) 2 4 2      B) 2 6 2      C) 6 2 2      D) 6 4 2      E) ไม่มีข้อใดถูก

# Final



## ลักษณะข้อสอบ Final

- การสอบ Midterm จะเก็บ 20 คะแนน
- ส่วนแรกเป็นการสอบปรนัย 5 ช้อยส์ จำนวน 50 ข้อ
- ส่วนที่สองเป็นการสอบเขียนโปรแกรมบนกระดาษ 3 ข้อ
- ระยะเวลาการสอบ 3 ชั่วโมง

**\*\*ข้อสอบ Final อย่าพลาดข้อสอบเขียนโปรแกรมบนกระดาษ**

# Final

## เนื้อหาในข้อสอบปี Final

บทที่ 07: String Processing

บทที่ 08: Basic Dict

บทที่ 09: Nested Structure

บทที่ 10: Tuple, Set, Dict

บทที่ 11: NumPy

บทที่ 12: Class & Object



## เนื้อหาในข้อสอบเขียนโปรแกรม

บทที่ 07: String Processing

บทที่ 11: NumPy

บทที่ 12: Class & Object

# Final

GET READY  
FRESHY



## ตัวอย่างข้อสอบ Final ส่วนปรนัย

```
class Order:  
    def __init__(self):  
        self.orderlines = []  
    def add(self, name, price):  
        self.orderlines.append( (name, price) ) # list of tuples  
    def total(self):  
        return sum([p for (n,p) in self.orderlines])  
    def __lt__(self, rhs):  
        return self.total() < rhs.total()  
def get_total(orders):  
    total = 0  
    for od in orders:  
        total += od.total()  
    return total  
o1 = Order(); o1.add("Congee",30); o1.add("Fried Rice",45); o1.add("Water",7)  
o2 = Order(); o2.add("Papaya Salad",40); o2.add("Congee",30)  
orders = [o1,o2,o1,o1]
```

37. หลังจากโปรแกรมข้างบนนี้ทำงาน ถ้าให้คำสั่ง `o2.orderlines[1][0]` จะคืนผลอะไร

- A) "Papaya Salad"
- B) "Fried Rice"
- C) "Water"
- D) "Congee"
- E) ไม่มีข้อใดถูก

# Final

## ตัวอย่างข้อสอบ Final ส่วนเขียนโปรแกรม

GET READY  
FRESHY

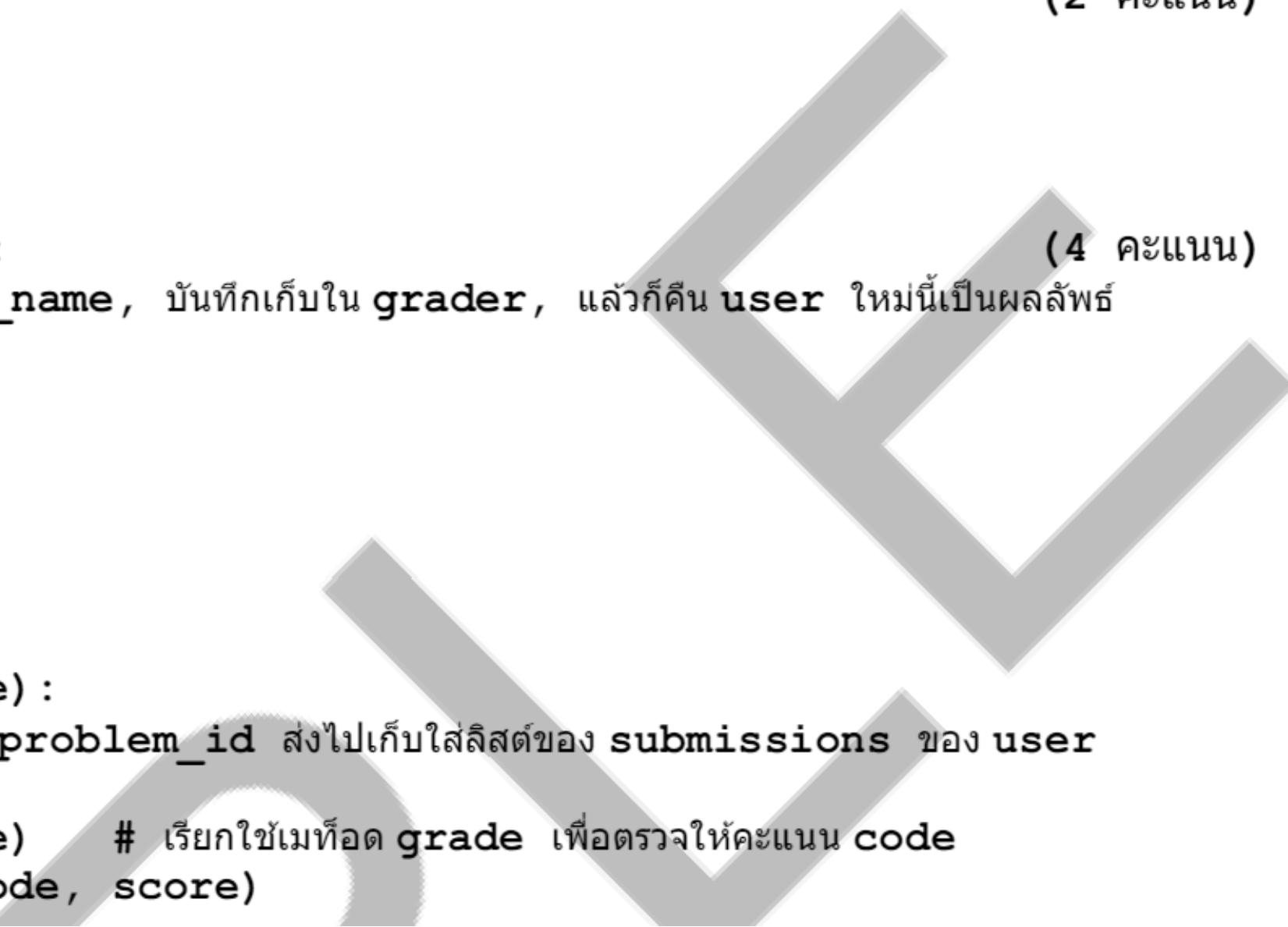


```
import random

class Grader:                                # ໄວสร้างօբเจկต์ของระบบ Grader ແນບ່າຍ ๆ
    def __init__(self):
        # ສ້າງທີ່ເກີນ user ຕ່າງໆ ຂອງຮະບນ grader

    def new_user(self, user_id, user_name):
        # ສ້າງ user ໃໝ່ທີ່ມີຮັສ user_id ແລະຊື່ user_name, ບັນທຶກໃນ grader, ແລ້ວກົດຕິ່ນ user ໃໝ່ນີ້ເປັນຜລັພີ້
        user =
        return user

    def submit(self, user, problem_id, code):
        # ຮັບການ submit ໂປຣແກຣມ code ຂອງໂຈທຍບໍ່ຢ່າງຫຼຸດ problem_id ສັງໄປເກີນໃສ່ລົສຕ່າງໆ ຂອງ user
        if user not in self.users: return
        score = self.grade(problem_id, code)      # ເຮັດໃຫ້ເມນີວິດ grade ເພື່ອຕຽບໃຫ້ຄະແນນ code
        user.save_submission(problem_id, code, score)
```



(2 ຄະແນນ)

(4 ຄະແນນ)

# สรุป Jun ท่ออักษร

GET READY  
FRESHY



บทที่ 01: Data Type & Expression	Grader 01	Midterm
บทที่ 02: Basic String & List		
บทที่ 03: Selection		
บทที่ 04: Repetition		
บทที่ 05: List Processing	Grader 02	Final
บทที่ 06: Function		
บทที่ 07: String Processing		
บทที่ 08: Basic Dict		
บทที่ 09: Nested Structure	Grader 03	Final
บทที่ 10: Tuple, Set, Dict		
บทที่ 11: NumPy		
บทที่ 12: Class & Object		



# Homework

การบ้านและงานในห้องเรียน

# Homework



## การบ้านเขียนโปรแกรม (10 คะแนน)

- การบ้านเขียนโปรแกรมจะมีทั้งหมด 5 ครั้ง รวมทั้งหมด 10 คะแนน
- การบ้านเขียนโปรแกรมจะกำผ่าน Google Colab
- ลักษณะการบ้านจะเป็นการเขียนฟังก์ชันต่างๆ ให้ทำงานตรงตามความต้องการของอาจารย์

**\*\*ห้ามลอกกันเด็ดขาด ไม่งั้นจะได้ 0 คะแนน**

# Homework

GET READY  
FRESHY

กบค.

## การบ้านที่ 2 ของปี 2566 เทอม 2 (ฟังก์ชันที่ 1)



```
# HW2_Genetics_Analysis
def get_acgt_count(data):
    """
```

(ฟังก์ชันนี้คิดเป็น 25% ของคะแนนการบ้านนี้)  
รับ data ซึ่งเป็นข้อมูลประเภทสตริง แทนข้อมูลทางพันธุกรรม  
- จะประกอบด้วยอักขระ A C G T เท่านั้น  
- จะมีอย่างละกี่ตัวก็ได้ (บางตัวจะไม่มีเลยก็ได้)  
- ความยาวของ data จะมากกว่า 0 เสมอ

ให้คืนค่าเป็นลิสต์ที่มีสมาชิกเป็นจำนวนเต็ม 4 ตัว ระบุจำนวนอักขระ A C G T ใน data

- ตัวแรกแทนจำนวนอักขระ A
- ตัวที่สองแทนจำนวนอักขระ C
- ตัวที่สามแทนจำนวนอักขระ G
- ตัวที่สี่แทนจำนวนอักขระ T

ตัวอย่าง CC

ให้ data เป็น "TGATT"

ถ้าเรียก get\_acgt\_count(data) ก็จะคืนค่าเป็น [1, 2, 1, 3] <-- โปรดสังเกตว่าเป็นลิสต์

ให้ data เป็น "AAGAA"

ถ้าเรียก get\_acgt\_count(data) ก็จะคืนค่าเป็น [4, 0, 1, 0]

"""

# Homework

GET READY  
FRESHY



## เฉลยการบ้านที่ 2 ของปี 2566 เทอม 2 (พังก์ชันที่ 1)



```
# HW2_Genetics_Analysis
def get_acgt_count(data):
    countA, countC, countG, countT = 0,0,0,0
    for char in data:
        if(char == 'A'): countA += 1
        elif(char == 'C'): countC += 1
        elif(char == 'G'): countG += 1
        elif(char == 'T'): countT += 1
    return [countA, countC, countG, countT]
```

# งานในห้องเรียน

## งานในห้องเรียน (10 คะแนน)

- งานในห้องเรียนแต่ละ Section จะไม่เหมือนกัน ขึ้นอยู่กับอาจารย์ที่สอน
- มักจะเป็นโจทย์เพิ่มเติมในห้องเรียน หรือเป็นการบ้านเล็กๆ
- บาง Section อาจจะสั่งให้ทำโจทย์ใน Grader ให้ครบ

**\*\*คะแนนส่วนนี้ควรเก็บเต็ม 10 คะแนน**



# งานในห้องเรียน

GET READY  
FRESHY



## งานในห้องเรียนชั้นที่ 11 (Section 2 ปี 2566 เทอม 2)

Complete the question set

1

ถ้าแต่ที่เรียนมา มีเรื่องไหนที่ยังไม่เข้าใจบ้าง และนิสิตแก้ปัญหานี้อย่างไร ได้ดู VDO ใน Playlists หัวข้อที่จะเรียนก่อนมาเข้าเรียนหรือไม่

1 point

Compose an answer

2

ทำแบบฝึกหัด 11-01 ข้อที่ 4 จาก <https://www.cp.eng.chula.ac.th/~somchai/python101/11-01.html>

1 point

Compose an answer

# งานในห้องเรียน



## งานในห้องเรียนชั้นที่ 4 (Section 5 ปี 2566 เทอม 2)

### คะแนนเกรดเฉลี่ย

เกรดเมื่อรายสิบปีก่อนไม่มีประจุ เป็นตัวอักษร A, B, C, D, F คะแนนของเกรดแต่ละตัวอักษรเป็นดังแสดงในตาราง

เกรด	A	B	C	D	F
คะแนน	4.0	3.0	2.0	1.0	0.0

ถ้าให้ทุกวิชา มีหน่วยกิตเท่ากันหมด เราสามารถคำนวณคะแนนเกรดเฉลี่ย จากสูตรข้างล่างนี้

$$\text{คะแนนเกรดเฉลี่ย} = \frac{\text{ผลรวมของคะแนนทุกวิชา}}{\text{จำนวนวิชา}}$$

### ข้อมูลนำเข้า

มี 1 บรรทัด เป็นตัวอักษรต่าง ๆ แทนเกรดของราย ๆ วิชา (A, B, C, D, F) แต่ละเกรดคั่นด้วยช่องว่างหนึ่งช่อง

### ข้อมูลส่งออก

คะแนนเกรดเฉลี่ยของข้อมูลนำเข้า แสดงเลขหลังจุดทศนิยม 2 ตำแหน่ง ด้วยฟังก์ชัน round( คะแนน, 2 )

# งานในห้องเรียน

GET READY  
FRESHY

Σ  
กวดค.

## งานในห้องเรียนชั้นที่ 4 (Section 5 ปี 2566 เทอม 2)

ตัวอย่าง

input (จากแป้นพิมพ์)	คะแนนเกรดเฉลี่ย	output (ทางจอกาพ)
A	4 / 1	4.0
A B	(4 + 3) / 2	3.5
A A B	(4 + 4 + 3) / 3	3.67
A B B	(4 + 3 + 3) / 3	3.33
A C B C	(4 + 2 + 3 + 2) / 3	2.75
F	0 / 1	0.0
B F	(3 + 0) / 2	1.5

# งานในห้องเรียน



## เฉลยงานในห้องเรียนชั้นที่ 4 (Section 5 ปี 2566 เทอม 2)

```
1 grades = input().strip().split()
2 sum = 0
3 for grade in grades:
4     if(grade == 'A'): sum += 4
5     elif(grade == 'B'): sum += 3
6     elif(grade == 'C'): sum += 2
7     elif(grade == 'D'): sum += 1
8 print(round(sum/len(grades) ,2))
```



ได้ 10/10 เด็มครับ :)

- TA Folk

Last graded on 19-02-2024 19:01:02 by Korraphat  
Pangvutthivanich

GET READY  
**FRESHY**



# How to Study

## เรียนอย่างไรให้ได้ A

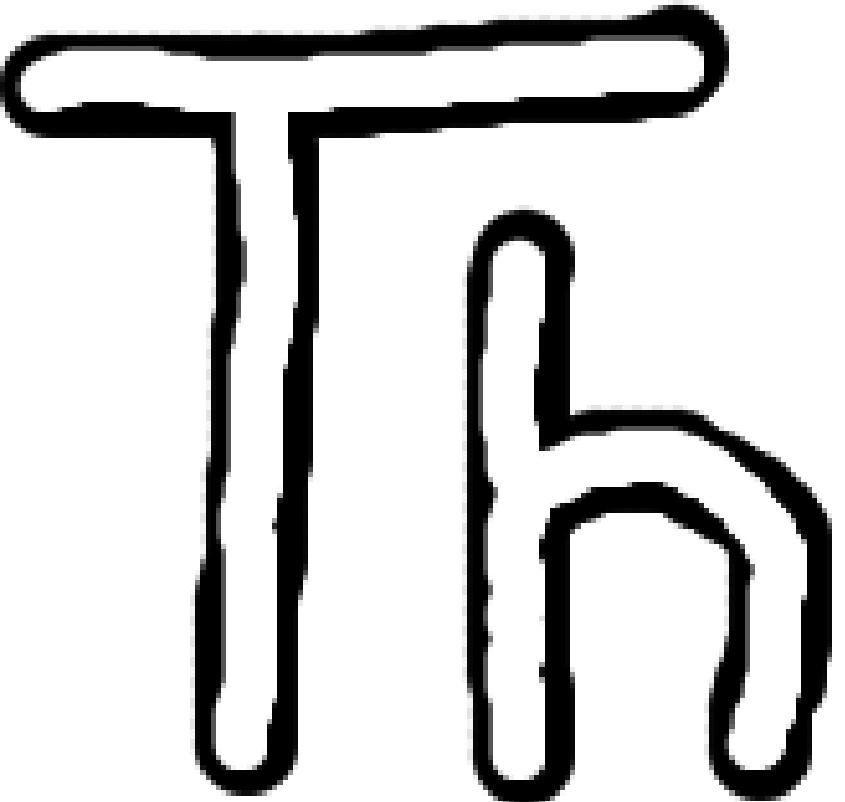
# โปรแกรมที่ใช้เรียน

GET READY  
FRESHY  
  
กวดค.



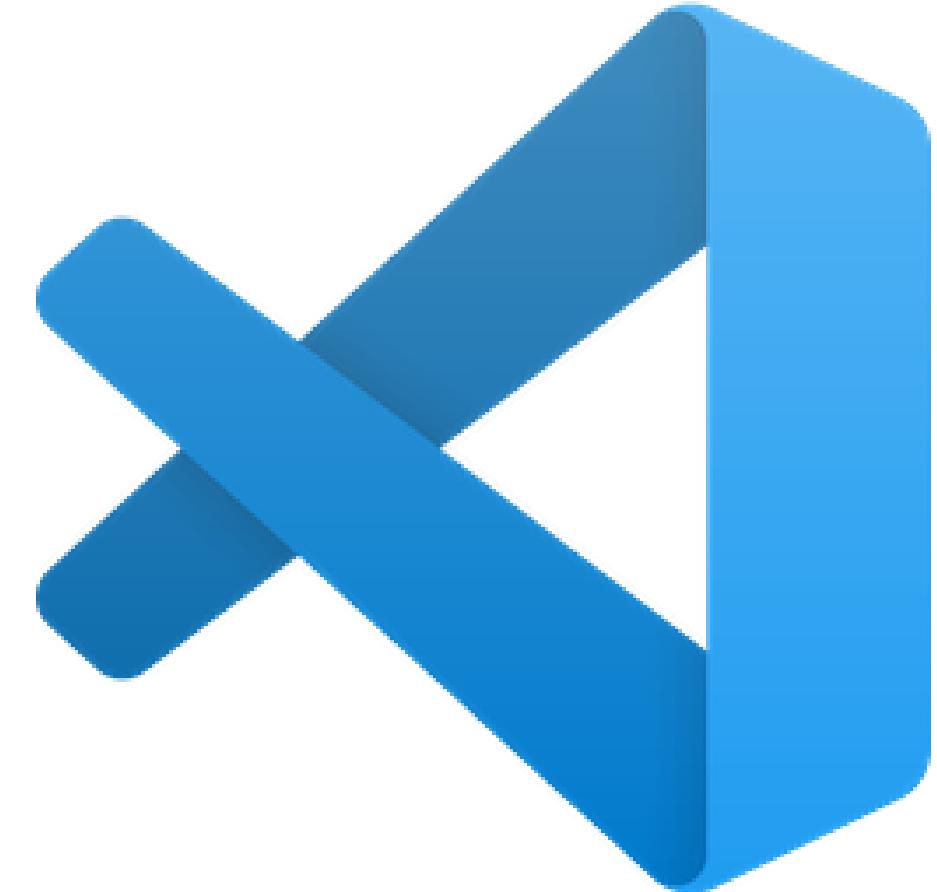
## Python

Download at [www.python.org](http://www.python.org)



## Thonny

Download at [www.thonny.org](http://www.thonny.org)



## vs Code

Download at  
[www.code.visualstudio.com](http://www.code.visualstudio.com)

(สำหรับภาคคอม)

# แหล่งการเรียนรู้

## 1 COM PROG Website

The screenshot shows the homepage of the Python101 website. At the top left is a red bell icon labeled "Python101". To its right are buttons for "ปี/เม็ด เมนู" and "Chapters ▾". The main content area features the Python logo and the text "python ๑๐๑" in large letters. Below this, it says "เริ่มเรียนเขียนโปรแกรม Python ๑๐๑" and "ภาควิชาวิศวกรรมคอมพิวเตอร์ จุฬาลงกรณ์มหาวิทยาลัย". A sidebar on the left lists chapters from 0 to 12, along with a "Books" section. A large text box at the bottom contains promotional text about the course's features and a QR code.

ปี/เม็ด เมนู Chapters ▾

Python101

> 0: Getting Started  
> 1: Data Type & Expression  
> 2: Basic String & List  
> 3: Selection  
> 4: Repetition  
> 5: List Processing  
> 6: Function  
> 7: String Processing  
> 8: Basic Dict  
> 9: Nested Structure  
> 10: Tuple, Set, Dict  
> 11: NumPy  
> 12: Class & Object  
Books

python ๑๐๑

เริ่มเรียนเขียนโปรแกรม Python ๑๐๑

ภาควิชาวิศวกรรมคอมพิวเตอร์ จุฬาลงกรณ์มหาวิทยาลัย

เรียนนี้รวมรวมบทเรียนการเขียนโปรแกรมขั้นพื้นฐานสำหรับผู้สนใจเริ่มเขียนโปรแกรม โดยใช้ภาษา Python เป็นเครื่องมือประกอบการเรียน ดังนั้นจึงไม่ได้นำเสนอภาษา Python อ้างอิงเดียว เราเลือกชุดค่าสั่งส่วนหนึ่งเพื่อประกอบการเรียนเขียนโปรแกรม แต่ก็เพียงพอที่จะเป็นพื้นฐานในการศึกษาเพิ่มเติมในระดับที่ซับซ้อนขึ้น

บทเรียนแบ่งเป็นหลาย ๆ บท แต่ละบทมีหลายหัวข้อ แต่ละหัวข้อประกอบด้วย VDO บรรยายเนื้อหา และตามด้วยแบบฝึกหัดให้ฝึกทำบนหน้าจอคอมพิวเตอร์ในหัวข้อนั้น โดยแบบฝึกหัดจะเป็นการเขียนโปรแกรม สามารถส่งผล เพื่อตรวจสอบโปรแกรมที่เขียนทำงานถูกต้องตามโจทย์หรือไม่ (ดูวิธีใช้งานได้ในบทที่ 0 หัวข้อ [DataCamp Demo](#))

ผู้สนใจสามารถอ่านและฝึกทำโจทย์เพิ่มเติมได้จากหนังสือในหัวข้อ [Books](#)

ขอให้สุขกับการเรียนสิ่งใหม่ และสุขกับการเขียนโปรแกรม Happy Coding จ้า...

[คณาจารย์ ดร.อดิศักดิ์ คงกระพัน](#)

© 2022 Copyright: ภาควิชาวิศวกรรมคอมพิวเตอร์ จุฬาลงกรณ์มหาวิทยาลัย

\*\*ใครอยากเก็บ A ต้องดูให้ครบถ้วนนะครับ

GET READY  
FRESHY



- คลิปวิดีโอสอนการเขียนโปรแกรมภาษา Python โดยอาจารย์สมชาย
- มีแบบฝึกหัดก้ายบกให้ทำ



# ແກ່ລົງກາຣເຮັຍບຸກ

## ① COM PROG Website

The screenshot shows a website interface for "Python101". On the left, there's a sidebar with a red sun icon and the text "Python101". Below it is a navigation menu:

- 0: Getting Started
  - 1: Introduction
  - 2: Software Installation
  - 3: DataCamp Demo.
  - 4: IDLE Demo.
  - 5: Thonny Demo.
- > 1: Data Type & Expression
- > 2: Basic String & List
- > 3: Selection
- > 4: Repetition
- > 5: List Processing
- > 6: Function
- > 7: String Processing
- > 8: Basic Dict
- > 9: Nested Structure
- > 10: Tuple, Set, Dict
- > 11: NumPy
- > 12: Class & Object
- > Books

The main content area has a blue header bar with "ປັດ/ປັບ ເມນຸ" and "Chapters ▾". Below it, the title "0: Getting Started" and subtitle "0-1: Introduction" are displayed. A video player window titled "101-2562-Python: 0.1 Introduction" shows a dark blue background with white text: "2110101 COMPUTER PROGRAMMING" and a play button icon. At the bottom of the video player, there's text in Thai: "ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย".

GET READY  
FRESHY



# แหล่งการเรียนรู้

## 1 COM PROG Website

### แบบฝึกหัด 4-2 ข้อที่ 1

ให้เขียนโปรแกรมรับจำนวนเต็ม  $n$  จาก input เพื่อแสดงจำนวนเต็มคี่ตั้งแต่ 1 ถึงจำนวนคี่ที่ไม่เกิน  $n$  (โดยใช้วงวน while)

script.py

```
1 n = int(input())
2 num = 1
3 while(num <= n):
4     print(num)
5     num += 2
6
```

IPython Shell

```
<script.py> output:
1
In [1]:
```

Great work!

Hint

Submit

Run



GET READY  
FRESHY

กบค.

# ແກ່ລົງກາຣເຮັຍບຽນ

## ② COM PROG Grader

The screenshot shows a digital gradebook interface with two assignments listed:

- Assignment 35: 05\_Nicknames**
  - Due Date: 5 days ago (06/03/24 20:40:29)
  - Score: 100.0 [PPPPPPPPPP]
  - Status: compiler msg, src, submissions (2), testcases
- Assignment 36: 05\_Back\_n\_Front**
  - Due Date: 2 days ago (09/03/24 23:22:31)
  - Score: 40.0 [PxxPPxxxxxP]
  - Status: compiler msg, src, submissions (2), testcases



- ແກ່ລົງຮວມໂຈກຍົກມພິວເຕອນ  
ຂອງรายวิชา COM PROG
- ບ້ອງໆຖຸກຄນະໄດ້ເຂົ້າໃຫ້ງານ  
Grader ເນື່ອເຮັມເຮັດວຽກ
- **ໃກຮອຍາກເກີບ A ແນະນຳມາກ່າວ  
ວ່າຕ້ອງກຳໂຈກຍີ້ໄດ້ 100  
ຄະແນນເຕີມທຸກຂ້ອນໃນ Grader**

# แหล่งการเรียนรู้

## ③ เลย์โทกซ์ Grader ของอาจารย์สมชาย

2110101 : เฉลย  
01\_Expr\_★\_Stirling\_Factorial

ภาควิชาวิศวกรรมคอมพิวเตอร์  
จุฬาลงกรณ์มหาวิทยาลัย  
๒๕๖๔

0:00 / 9:25

2110101: เฉลย 01\_Expr\_★\_Stirling\_Factorial

s somchaiSPJ ผู้ติดตาม 1.26 หมื่น คน

46 ติดตามแล้ว 46 แชร์ ดาวน์โหลด คลิป ...

2110101 Grader Practices: Solutions somchaiSPJ - 1 / 79

2110101: เฉลย 01\_Expr\_★\_Stirling\_Factorial somchaiSPJ 9:26

2110101: เฉลย 01\_Expr\_★\_Quadratic\_Root somchaiSPJ 6:31

2110101: เฉลย 01\_Expr\_★\_An\_Expression somchaiSPJ 10:52

2110101: เฉลย 01\_Expr\_★\_Body\_Surface\_Area somchaiSPJ 5:24

2110101: เฉลย 01\_Expr\_★★\_Duration somchaiSPJ 18:47

2110101: เฉลย 02\_StrLst\_★\_CitizenID somchaiSPJ 15:18

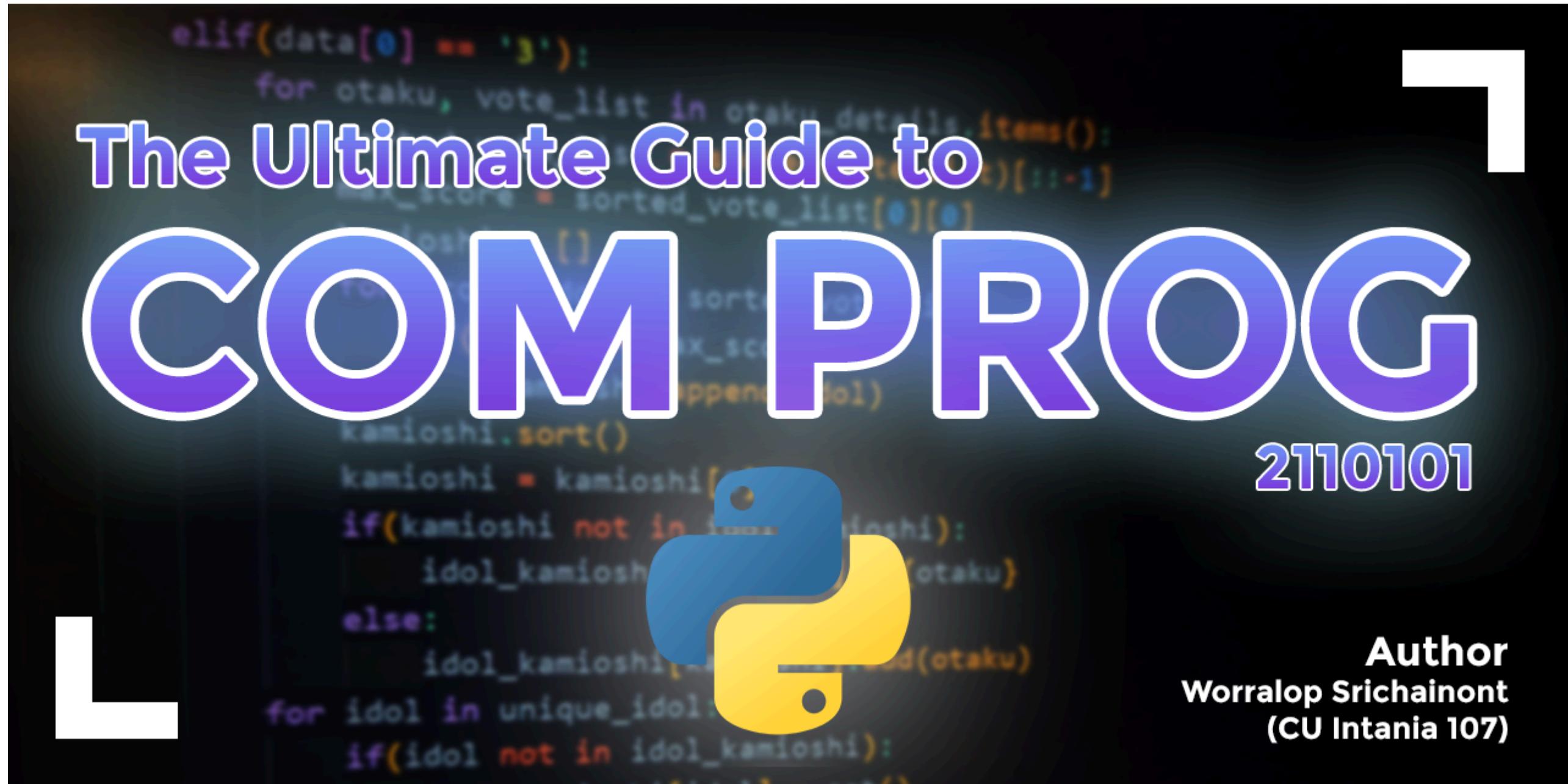
2110101: เฉลย 02\_StrLst\_★\_Arabic\_Numerals somchaiSPJ 3:58

GET READY  
FRESHY



# ॥ការងារ॥ រៀលប្រតិបត្តិ

④ COM PROG GitHub



GET READY  
FRESHY



# แหล่งการเรียนบัญชี



## ④ COM PROG GitHub

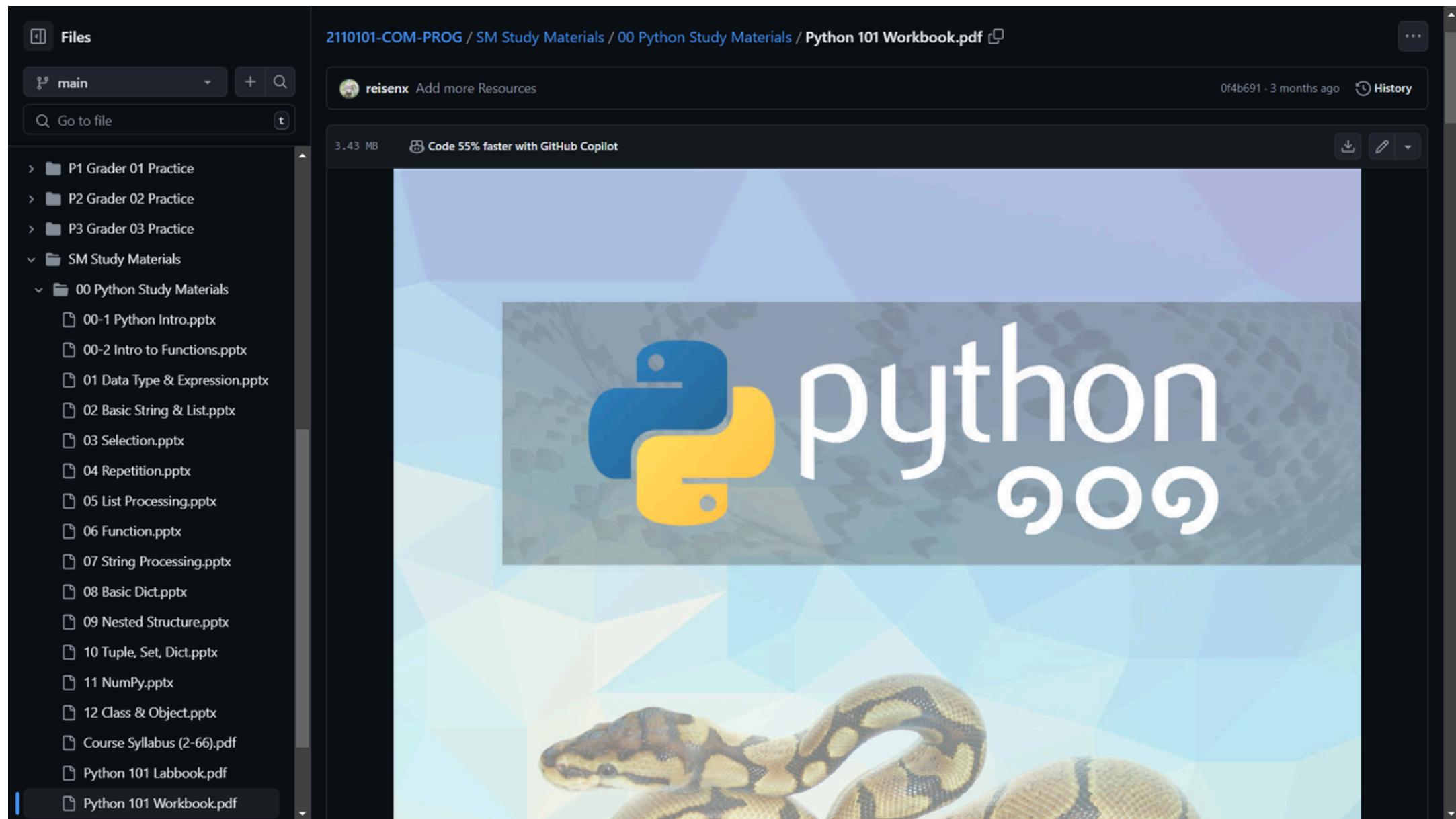
### COM PROG GitHub มีอะไรบ้างนะ

- รายละเอียดรายวิชา และเอกสารประกอบการเรียน พร้อมเรียนกันที่ไฟล์แบบฝึกหัด Grader **ทุกข้อ** และข้อสอบเก่าย้อนหลังถึงปี 2565
- เฉลยอย่างละเอียดแบบฝึกหัด Grader และข้อสอบเก่าย้อนหลัง**ทุกข้อ** (รับประกันเต็ม 100 คะแนน)
- เฉลยตัวอย่างข้อสอบ Midterm และ Final อย่างละเอียด
- Cheat Sheet สำหรับเข้าสอบ Grader แต่ละครั้ง

**\*\*จัดทำและพัฒนาโดย นายวรลักษย์ ศรีชัยบันท (P'Bloom #2 CP)**

# ॥ກລົງກາຣເຮັດນຸ່ງ

## ④ COM PROG GitHub



ຮັບຮວມເອກສາຣປະກອບກາຣເຮັດ

GET READY  
**FRESHY**



# แหล่งการเรียนรู้

## 4 COM PROG GitHub

The screenshot shows a GitHub repository interface. On the left, the file structure is visible, including a folder named 'G66 Grader 2566' which contains several PDF files related to quizzes and test cases. The main area displays a PDF document titled '2566\_1\_Quiz\_3\_3.pdf'. The document contains a graph problem with the following text:

ตัวอย่างแผนที่เดิน  
6  
NewMarineford PunkHazard 5  
WanoKuni Sphinx 10  
Wholecakelsland WanoKuni 3  
PunkHazard Applenineland 7  
Applenineland NewMarineford 1  
Zou Wholecakelsland 9

Below the text is a graph diagram with nodes labeled: NewMarineford, PunkHazard, WanoKuni, Wholecakelsland, Applenineland, Zou, and Sphinx. The edges and their weights are: (NewMarineford, PunkHazard) = 5, (PunkHazard, WanoKuni) = 3, (WanoKuni, Wholecakelsland) = 10, (Wholecakelsland, Applenineland) = 9, (Applenineland, NewMarineford) = 1, (Applenineland, Zou) = 7, and (Zou, Sphinx) = 6.

At the bottom, there is a table comparing input and output for different routes:

Input (จากแป้นพิมพ์)	Output (ทางจอภาพ)
6	12 Route not found
NewMarineford PunkHazard 5	
WanoKuni Sphinx 10	6 Route not found
Wholecakelsland WanoKuni 3	
PunkHazard Applenineland 7	
Applenineland NewMarineford 1	
Zou Wholecakelsland 9	

รวมโจทย์ Grader และข้อสอบเก่าย้อนหลัง

GET READY  
FRESHY



# ແກ່ລົງກາຣເຮັຍບຽນ

## ④ COM PROG GitHub



The screenshot shows a GitHub code editor interface. The repository path is 2110101-COM-PROG / G66 Grader 2566 / 2566\_1\_Quiz\_2\_2.py. The code is a Python script named 2566\_1\_Quiz\_2\_2.py. The script reads student details from a file and stores them in a dictionary. The code is as follows:

```
4 # Input filename
5 filename = input().strip()
6
7 # Open a file
8 file = open(filename, "r")
9
10 # Read student details in a file
11 # Each line in the file will be in this format
12 # [ID],[Gender],[Faculty code],[Section],[Professor]
13 # We will read and collect the data in the dictionary in this format
14 # professor = {Professor:[[List of sections], Female, Male]}
15 professor = {}
16 for line in file:
17     ID,gender,faculty,sec,prof = line.strip().split(',')
18     sec = int(sec)
19     # Add a new data, if that professor data is not exists yet
20     if(prof not in professor):
21         if(gender == 'F'):
22             professor[prof] = [[sec],1,0]
23         elif(gender == 'M'):
24             professor[prof] = [[sec],0,1]
25     # Edit data, if that professor data is already exists
26     else:
27         # Check gender
```

GET READY  
FRESHY



ເລຍອຢ່າງລະເວີດຂ້ອສົບ Grader ແລະຂ້ອສົບເກົ່າທຸກຂ້ອ

GET READY  
FRESHY



# How to get A

## Step 1 เข้าเรียนทุกคาบ



สำหรับผู้ที่**ไม่มีพื้นฐานเลย** แนะนำให้เข้าเรียนทุกคาบ เพราะว่า

- เข้าไปฟังอาจารย์สอนเนื้อหา
- ฝึกกำจดอยู่ใน Grader ในคาบ
- ถ้าเราสองสัย หรือกำจดยังไม่ได้ เราสามารถสอบถามอาจารย์ และ TA ประจำห้องได้ทันที

ปล. วิชานี้เรียนออนไลน์ได้ด้วยนะ

ปกติเราจะเรียนวิชานี้กันที่ศูนย์คอมกับนะครับ

# How to get A



## Step 2 นบทวนบทเรียนกับคลิปอาจารย์สมชาย

The screenshot shows a video player interface. At the top left is a red bell icon labeled "Python101". To its right are buttons for "ปิด/เปิด เมนู" and "Chapters ▾". The main content area displays the title "0: Getting Started" and "0-1: Introduction". Below this is a video thumbnail with the text "101-2562-Python: 0.1 Introduction" and a play button icon. The video title "2110101 COMPUTER PROGRAMMING" is prominently displayed in large white letters. At the bottom of the video frame, there is text in Thai: "ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย". On the far left of the video player, there is a sidebar with a list of chapters:

- 0: Getting Started
- 1: Introduction
- 2: Software Installation
- 3: DataCamp Demo.
- 4: IDLE Demo.
- 5: Thonny Demo.
- > 1: Data Type & Expression
- > 2: Basic String & List
- > 3: Selection
- > 4: Repetition
- > 5: List Processing
- > 6: Function
- > 7: String Processing
- > 8: Basic Dict
- > 9: Nested Structure
- > 10: Tuple, Set, Dict
- > 11: NumPy
- > 12: Class & Object
- > Books

- เข้า COM PROG Website เพื่อไปเรียนการเขียนโปรแกรม
- เมื่อเรียนเสร็จให้ลองทำแบบฝึกหัดข้างล่างเพื่อทดสอบความเข้าใจของตนเอง
- อย่าดองคลิปไว้ดูตอนใกล้ๆสอบ มันไม่ช่วยอะไรมากเลยจริงๆ**

# How to get A



## Step 3 ฝึกฝนทำโจทย์ใน Grader

The screenshot shows a digital interface with a blue header bar containing a house icon and the text 'MAIN'. Below the header, there are two assignment entries:

**Assignment 35: 05\_Nicknames ★**  
Status: 100.0 [PPPPPPPPPP]  
Submissions: 2 / Testcases: 2  
Last Submission: 5 days ago (06/03/24 20:40:29)  
Actions: Edit, Read

**Assignment 36: 05\_Back\_n\_Front ★**  
Status: 40.0 [PxxPPxxxxP]  
Submissions: 2 / Testcases: 2  
Last Submission: 2 days ago (09/03/24 23:22:31)  
Actions: Edit, Read

- ฝึกฝนทำโจทย์ของบทที่เรียนใน COM PROG Grader
- **ถ้าอยากได้ A จริงๆ ต้องทำโจทย์ให้ได้ 100 คะแนนเต็ม ครับ**



# How to get A

## Step 3 ฝึกฝนทำโจทย์ใน Grader

**Q:** แล้วถ้าเกิดทำโจทย์ใน Grader ไม่ได้ขึ้นมา ทำยังไงดีล่ะ

**A:** สามารถแก้ปัญหานี้ได้หลายวิธี

- ดูคลิปเฉลยโจทย์ Grader ของอาจารย์สมชาย
- ดูเฉลยใน COM PROG GitHub
- สอดถามอาจารย์ หรือ TA ใน Discord ของรายวิชา
- สอดถามเพื่อนๆที่เก่งวิชานี้

**\*\*ก่อนดำเนินการพยายามทำโจทย์ข้อนี้ให้สุดความสามารถก่อน**

# How to get A

GET READY  
FRESHY



## Step 3 ฝึกฝนทำโจทย์ใน Grader

The screenshot shows a Discord channel named "grader-help". The channel has several messages. One message titled "11\_Lower\_than\_Mean" contains a Python code snippet for reading data from input and calculating the mean. Another message shows a test case for a bidding system. There are also messages from TA Folk providing explanations and solutions to problems.

```
import numpy as np

def read_data():
    w = [float(e) for e in input().split()]
    weight = np.array(w)
    n = int(input())

    return weight, n
```

สอบถามอาจารย์หรือ TA ใน Discord ของรายวิชา

# How to get A

GET READY  
FRESHY



## Step 4 ฝึกทำข้อสอบเก่าแบบจับเวลา

The screenshot shows a GitHub repository interface. On the left, the file structure is visible, including a folder for 'G66 Grader 2566' which contains several PDF files for quizzes. The main area displays a PDF document titled '2110101-COM-PROG / SM Study Materials / 01 Grader Problems / G66 Grader 2566 / 2566\_1\_Quiz\_3\_3.pdf'. The document contains a graph problem and a table for input and output.

**Graph Problem:**

- 节点: NewMarineford, PunkHazard, WanoKuni, Wholecakelsland, Appleninelsland, Zou, Sphinx.
- 边及权重:

  - NewMarineford → PunkHazard (5)
  - NewMarineford → Appleninelsland (1)
  - PunkHazard → Wholecakelsland (3)
  - Wholecakelsland → Zou (9)
  - Wholecakelsland → Sphinx (10)
  - Appleninelsland → Zou (7)
  - Appleninelsland → NewMarineford (1)
  - Zou → Sphinx (6)

**Table:**

Input (จากเป็นพิมพ์)	Output (ทางภาษาพ.)
6	12
NewMarineford PunkHazard 5	Route not found
WanoKuni Sphinx 10	6
Wholecakelsland WanoKuni 3	Route not found
PunkHazard Appleninelsland 7	Route not found
Appleninelsland NewMarineford 1	Route not found

- หาข้อสอบเก่าย้อนหลังได้ที่ COM PROG GitHub
- Download ข้อสอบเก่า 3 ข้อ และลองทำแบบจับเวลา เมื่อสอบจริงๆ 90 นาที

# How to get A



## Step 5 ส่งการบ้านให้ครู และตั้งใจสอบ

รายการ	คะแนน
สอบเก็บคะแนน Grader	45
สอบกลางภาค	15
สอบปลายภาค	20
การบ้านเขียนโปรแกรม	10
งานในห้องเรียน	10
<b>รวม</b>	<b>100</b>

คะแนนส่วนนี้สำคัญที่สุด  
ส่วนนี้เก็บคะแนนง่าย

แนะนำให้เก็บเต็ม

# How to get A

GET READY  
FRESHY



- Step 1** เข้าเรียนทุกคาบ
- Step 2** นุนหวนบทเรียนกับคลิปอาจารย์สมชาย
- Step 3** ฝึกฝนทำโจทย์ใน Grader
- Step 4** ฝึกทำข้อสอบเก่าแบบจับเวลา
- Step 5** ส่งการบ้านให้ครู และตั้งใจสอบ

GET READY  
**FRESHY**



# Statistics

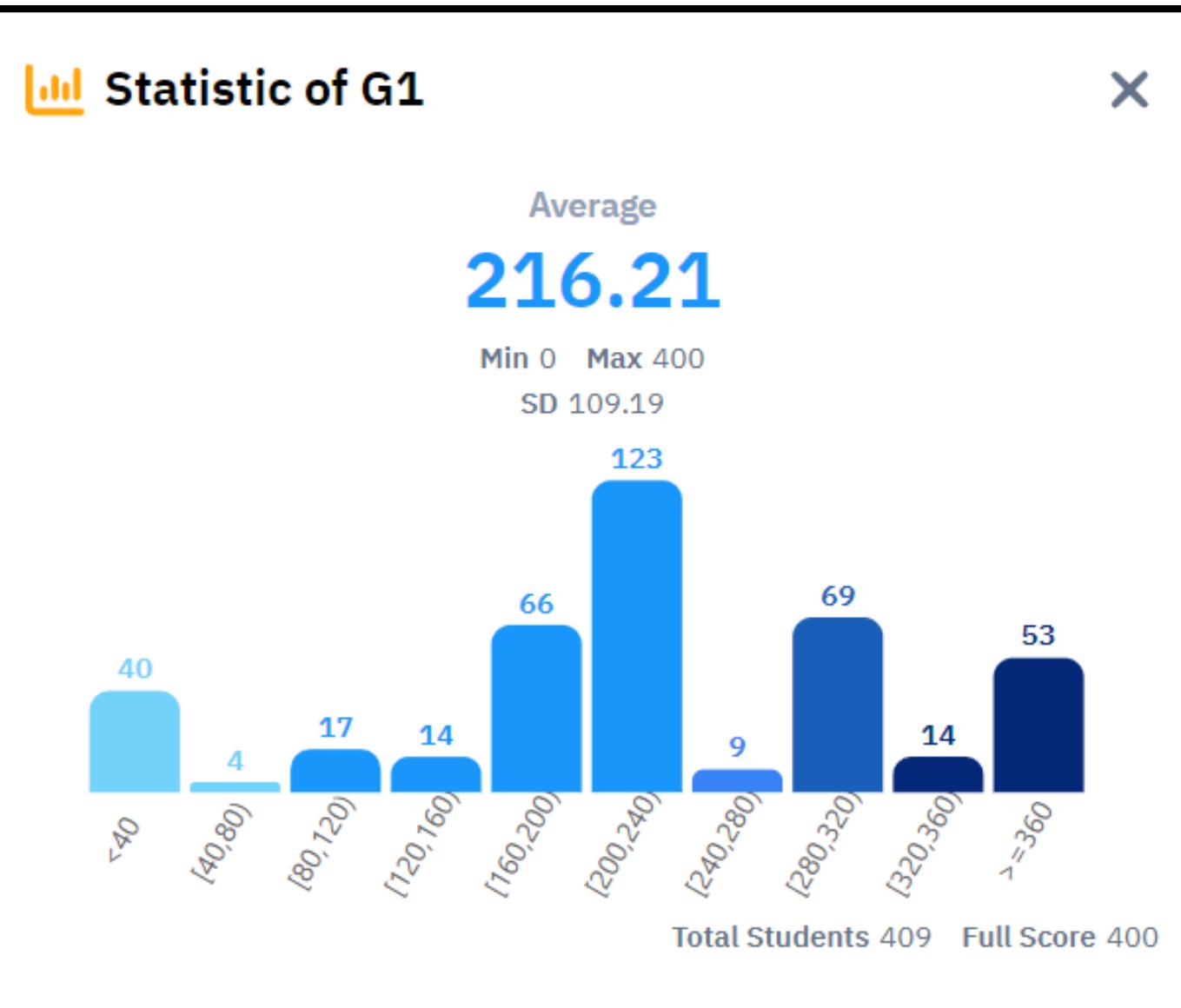
สถิติคณิตศาสตร์รายวิชานี้

# สถิติคะแนน

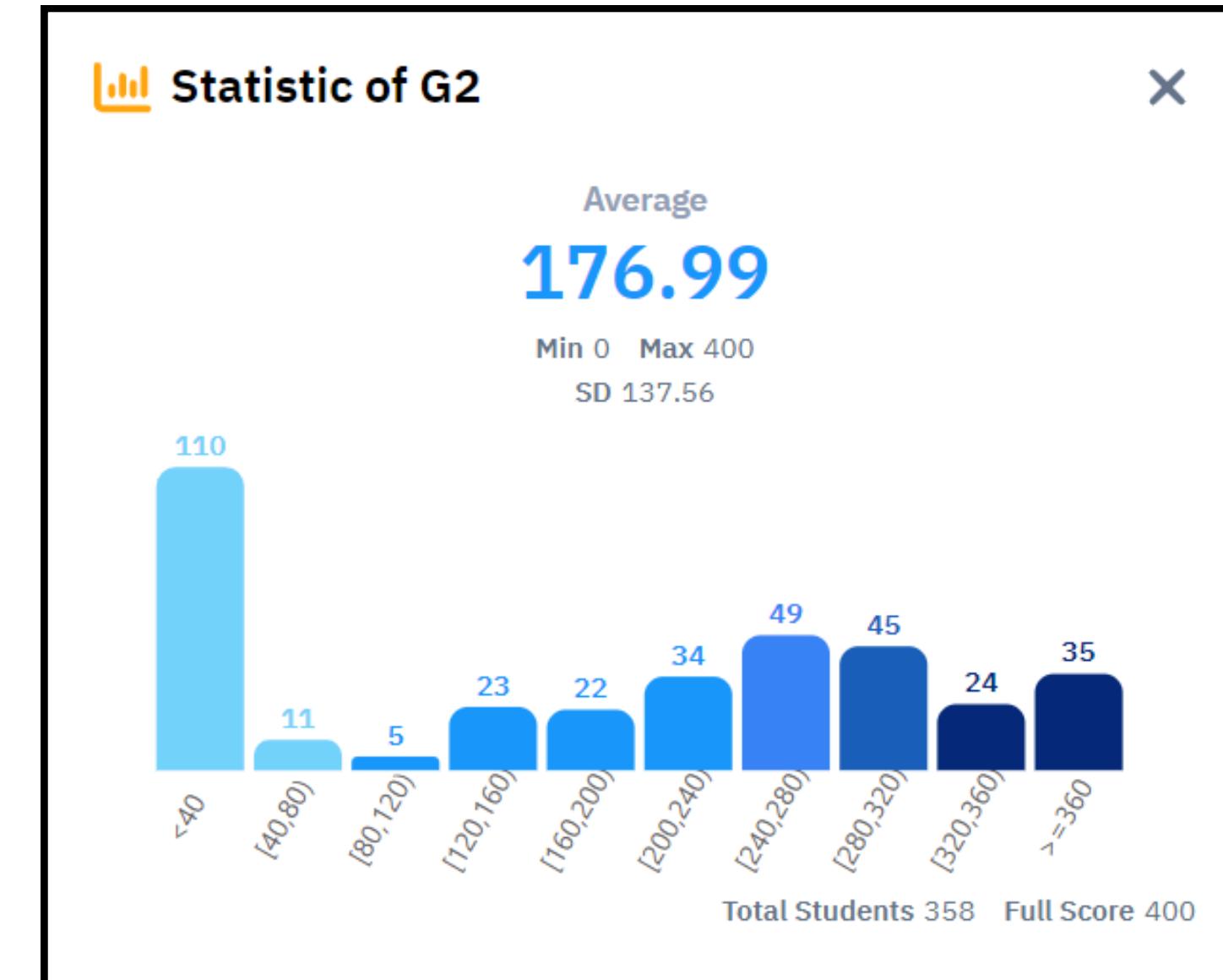
## สถิติคะแนน COM PROG ปี 2566 เทอม 1

GET READY  
FRESHY

กบค.



Grader 01



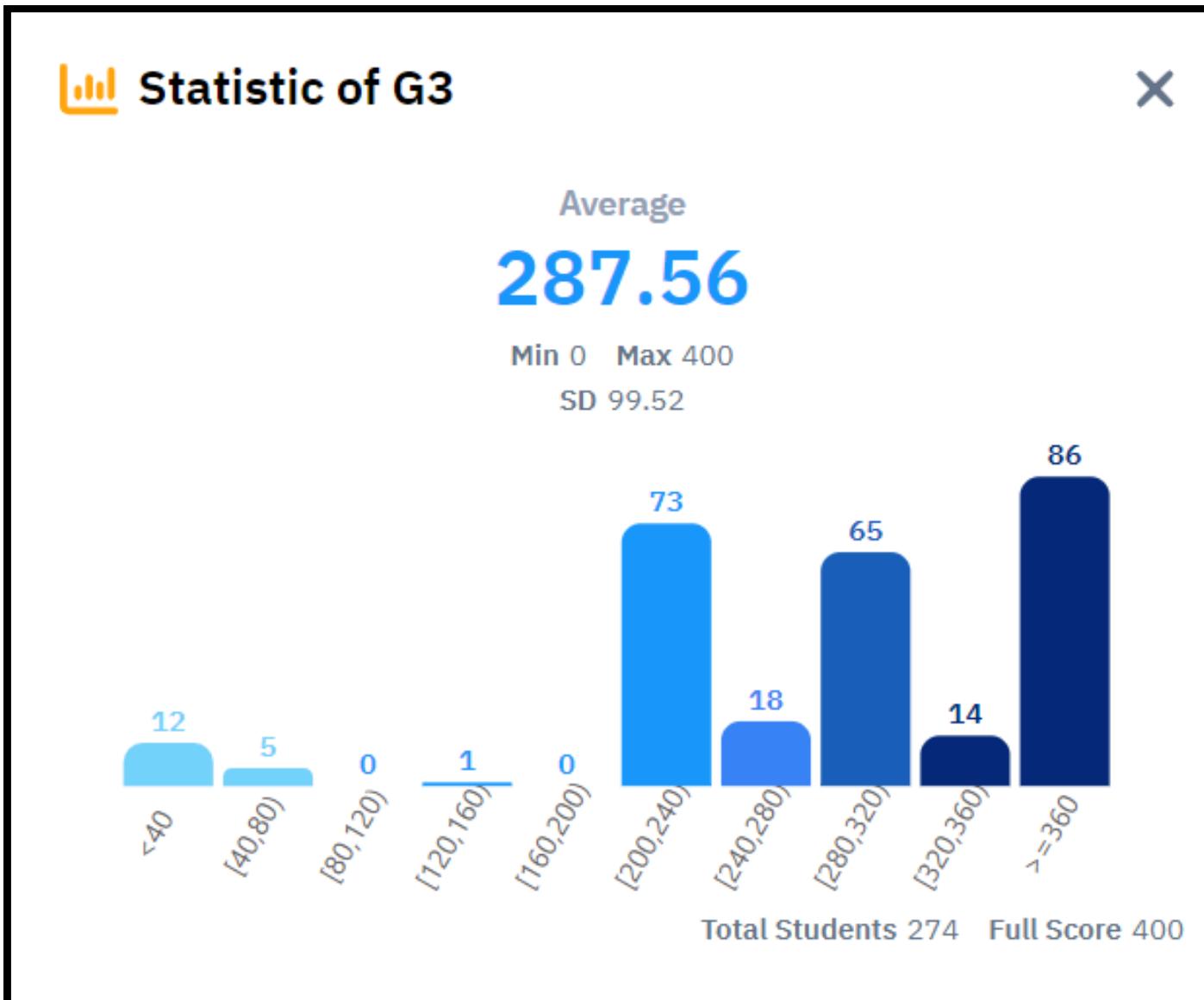
Grader 02

# ສັດຕິຄະແນນ

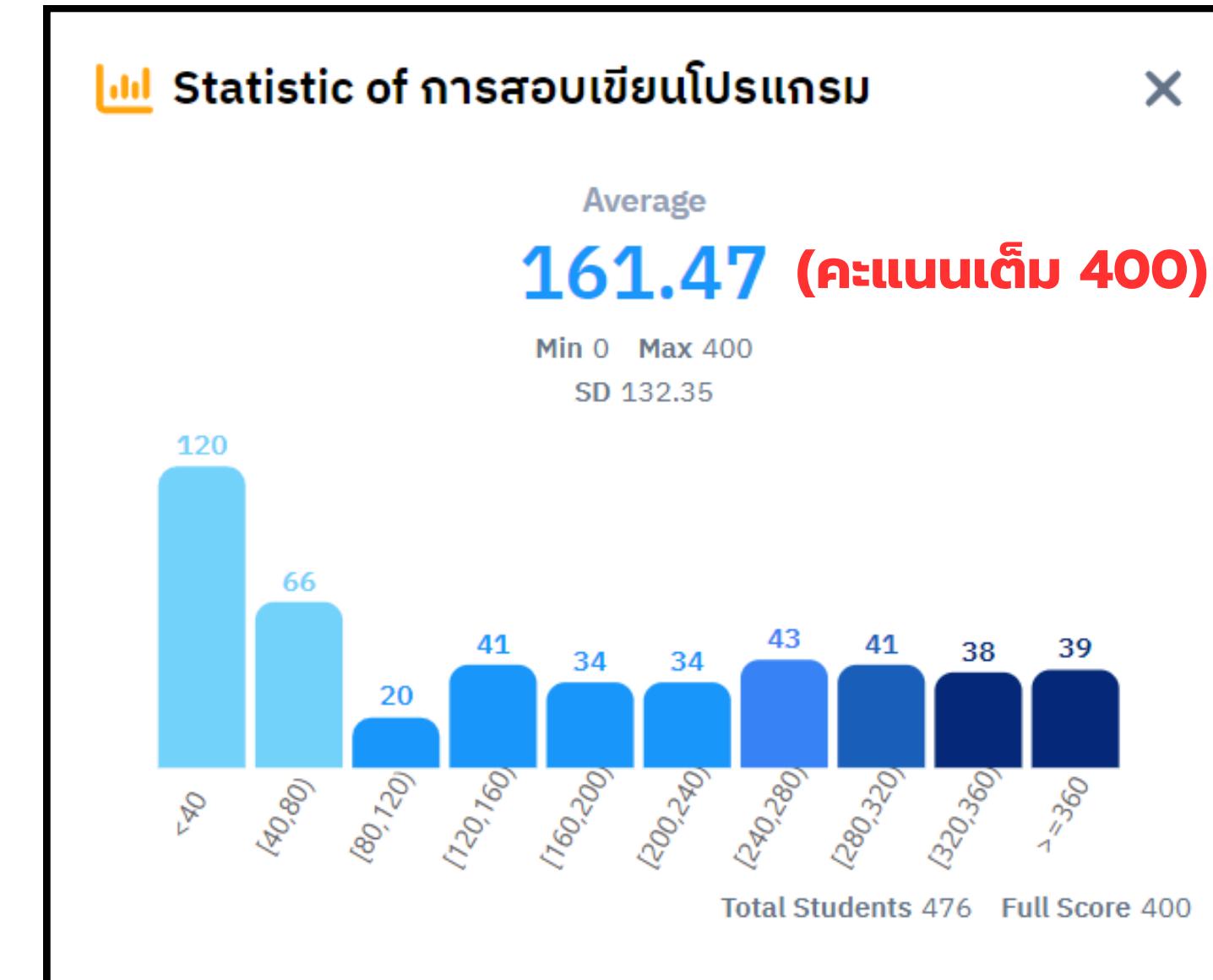
GET READY  
FRESHY

กວດ.

## ສັດຕິຄະແນນ COM PROG ජິງ 2566 ເກອມ 1



Grader 03



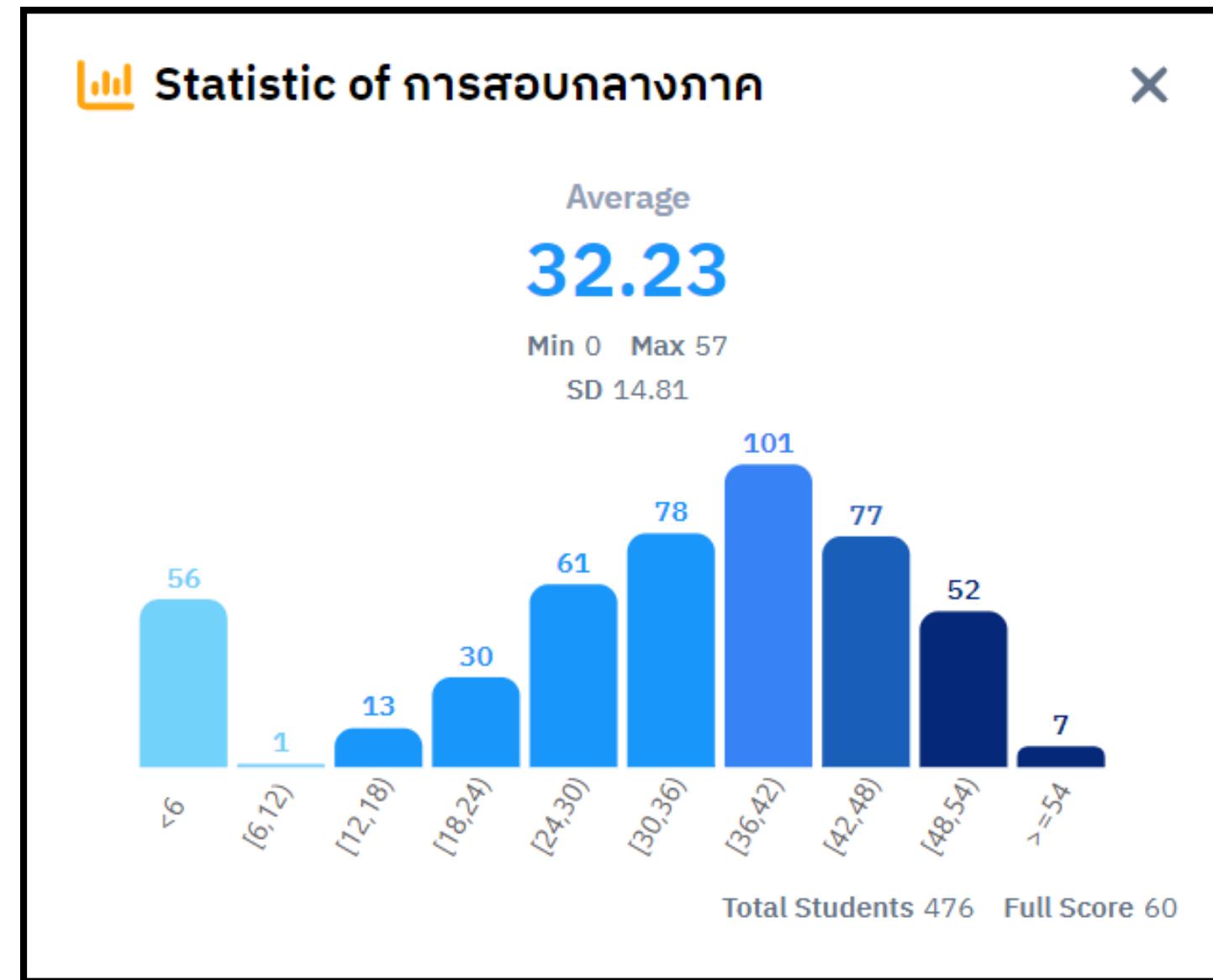
ຄະແນນຮຽນ Grader

# สถิติคะแนน

## สถิติคะแนน COM PROG ปี 2566 เทอม 1

GET READY  
FRESHY

กบค.



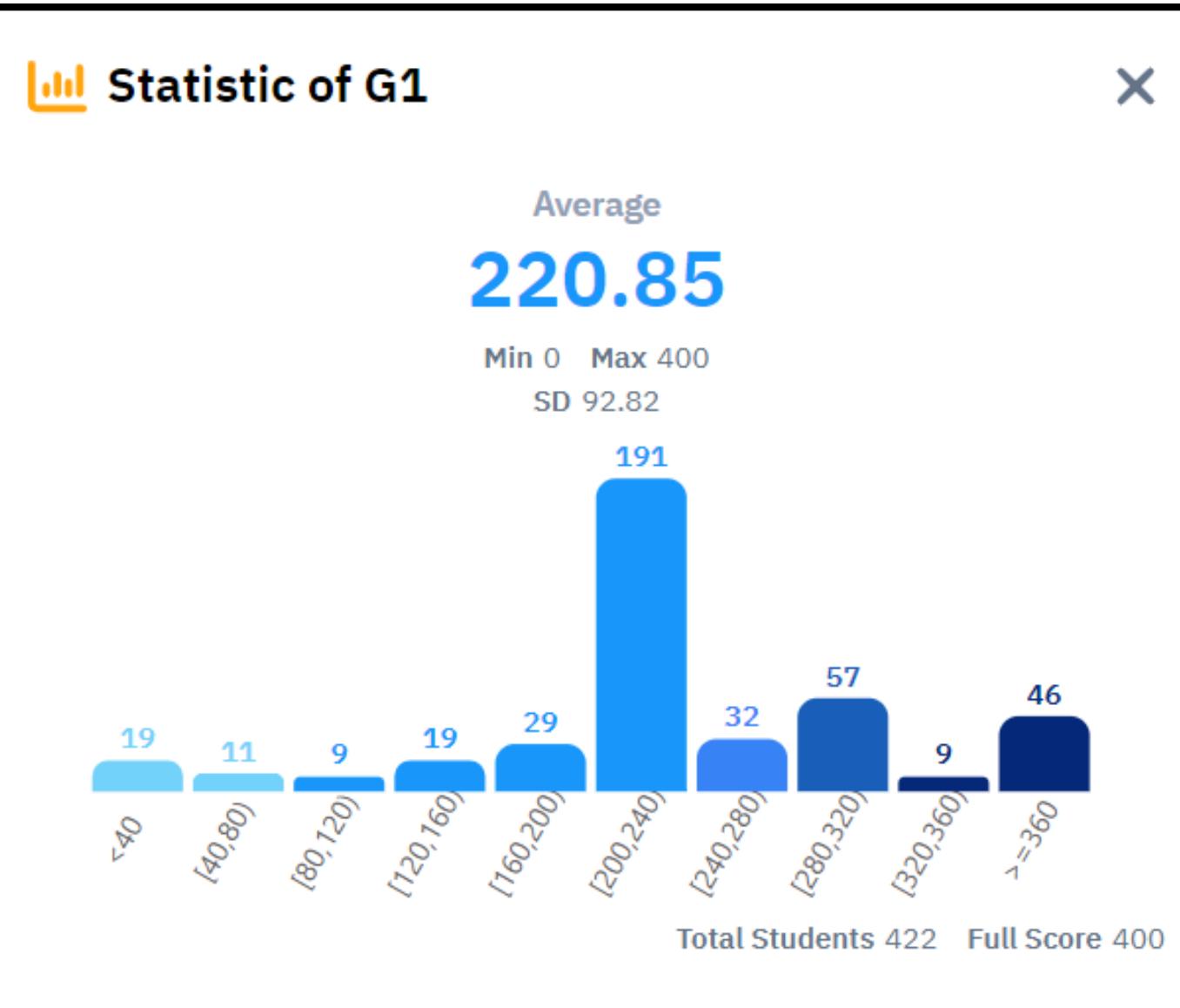
คะแนนสอบ Midterm

# ສັດຕິຄະແນນ

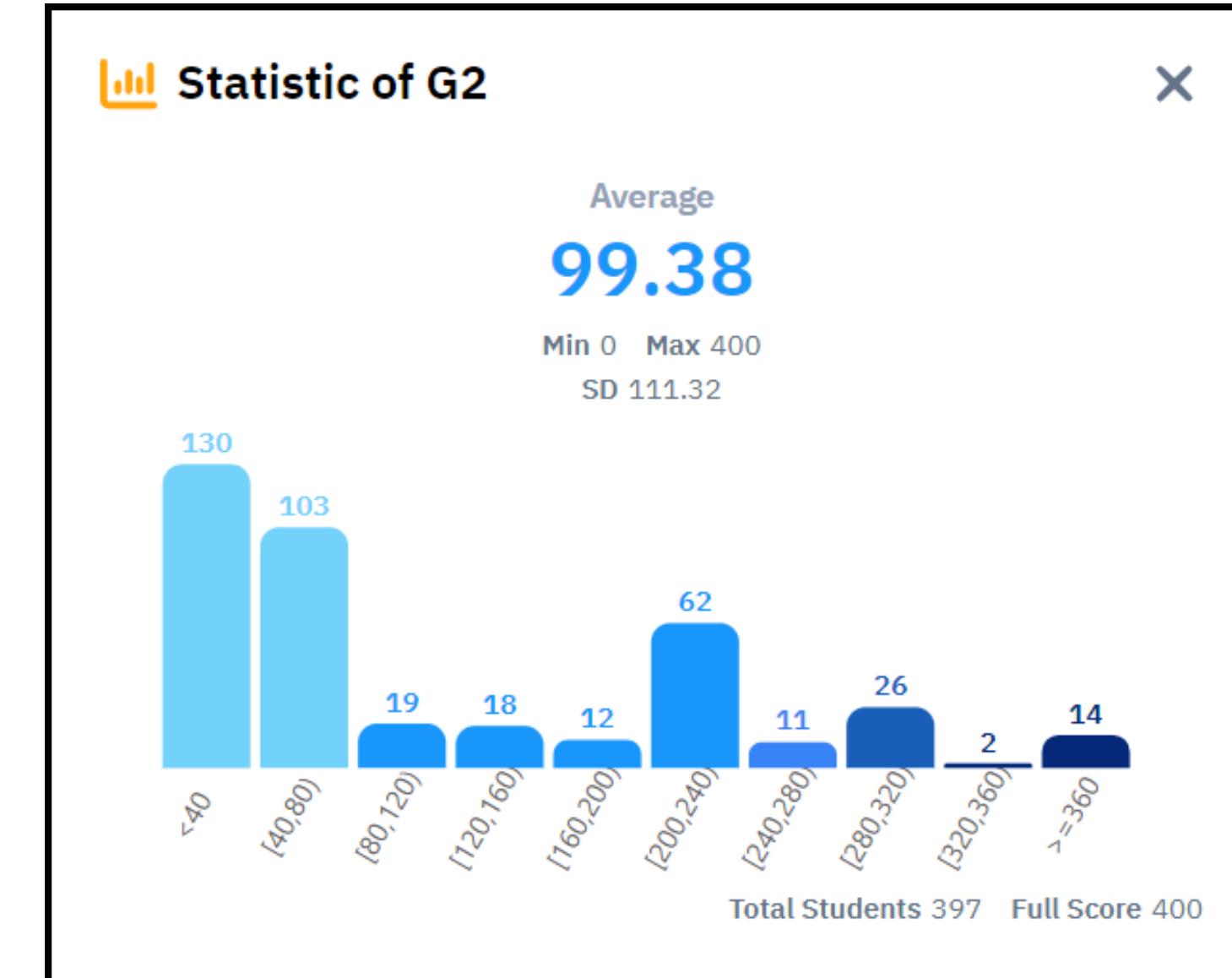
GET READY  
FRESHY

กວດ.

## ສັດຕິຄະແນນ COM PROG ປີ 2566 ເກອມ 2



Grader 01



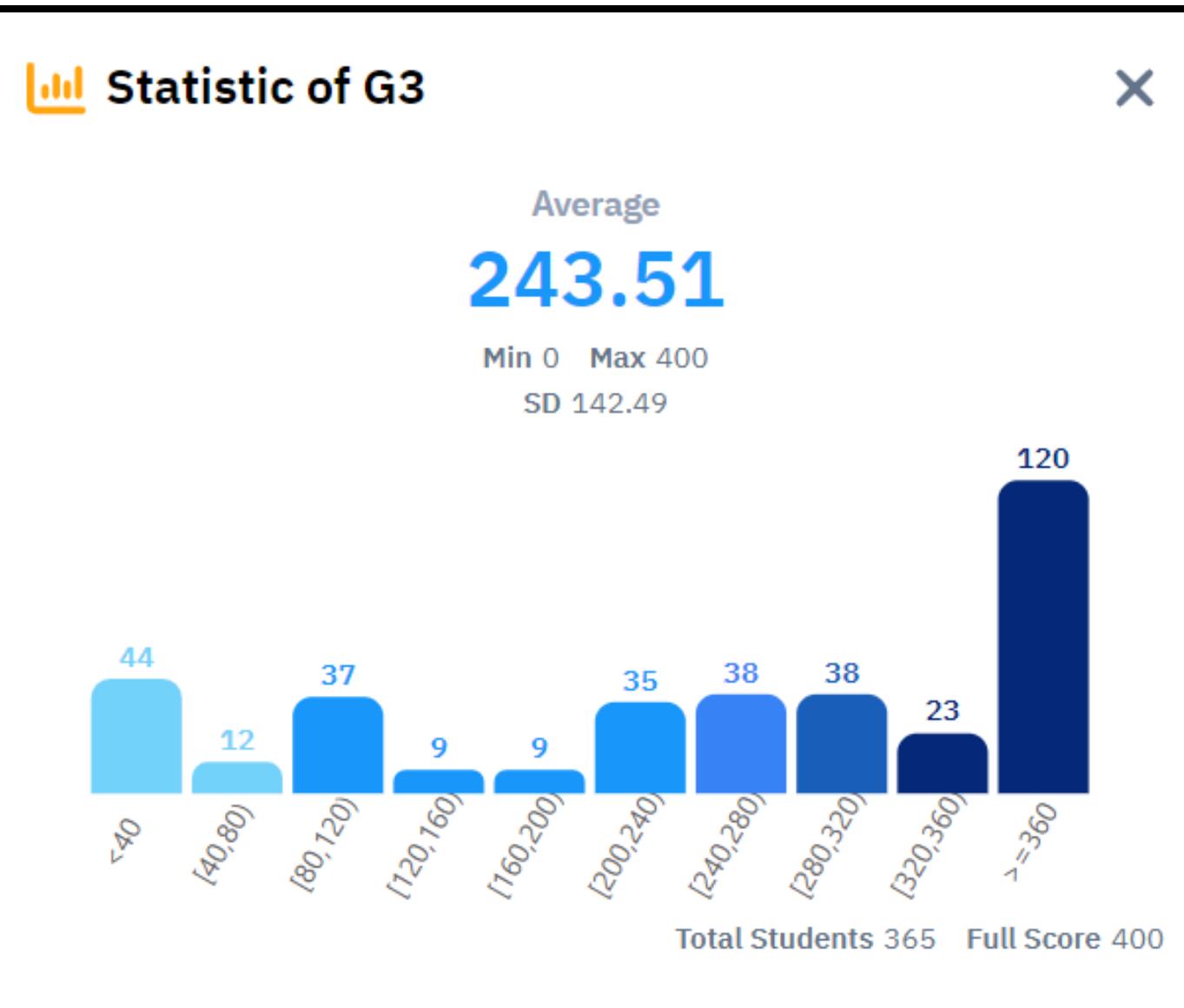
Grader 02

# ສັດຕິຄະແນນ

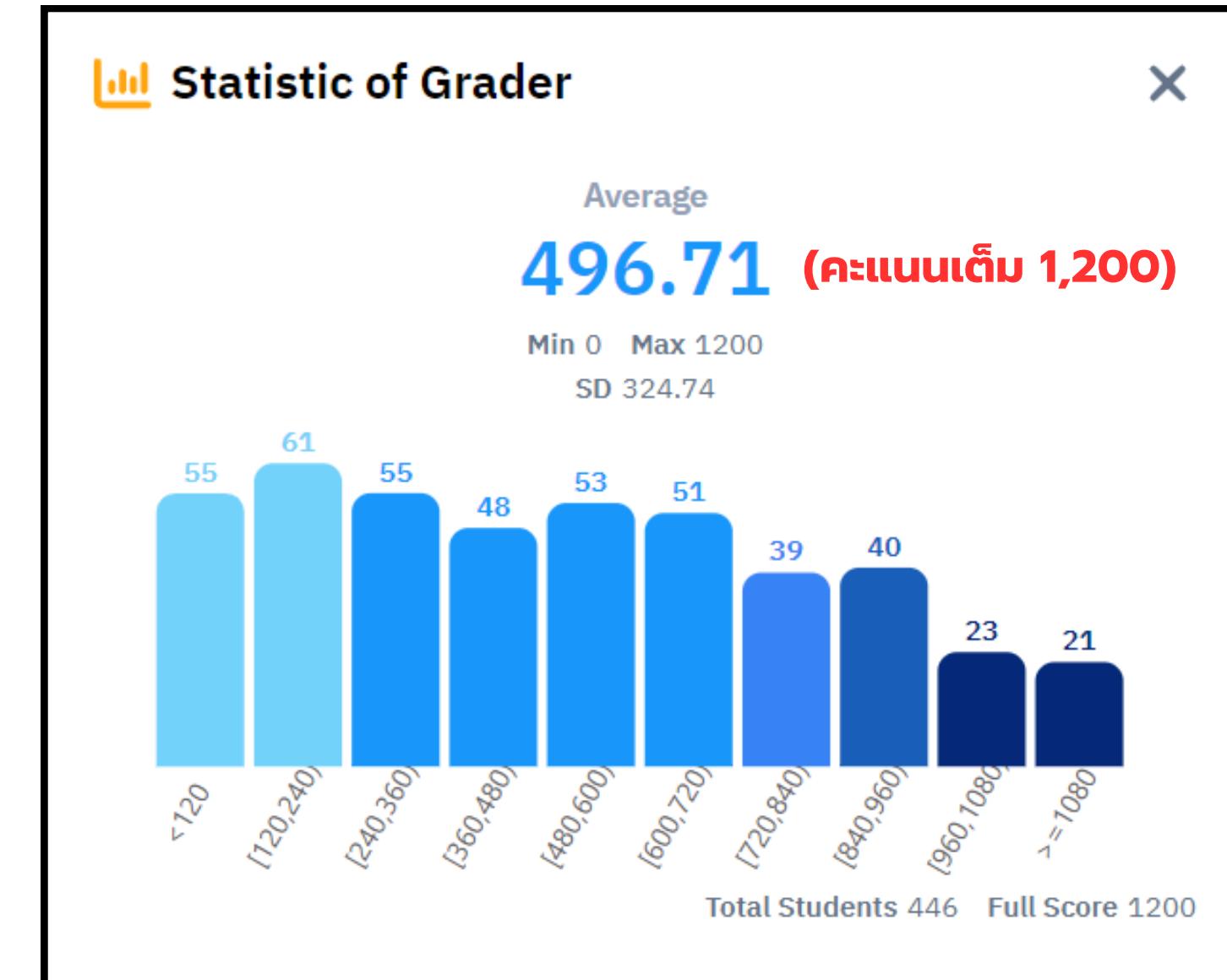
GET READY  
FRESHY

กວດ.

## ສັດຕິຄະແນນ COM PROG ປີ 2566 ເກອມ 2



Grader 03



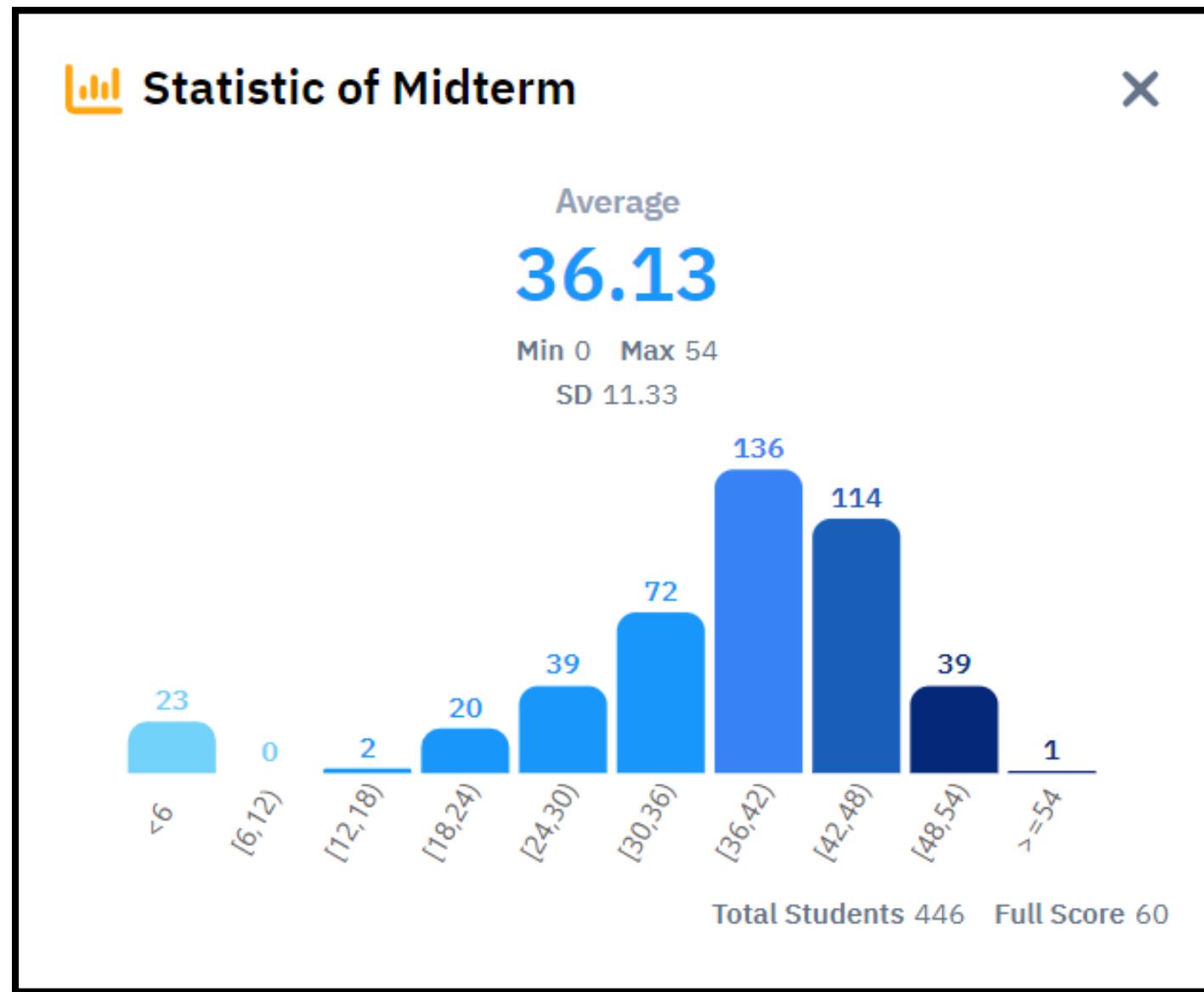
ຂະແນນຮຽນ Grader

# สถิติคะแนน

## สถิติคะแนน COM PROG ปี 2566 เทอม 2

GET READY  
FRESHY

กบก.



คะแนนสอบ Midterm

# เกี่ยบสติคະແນນ



## เกี่ยบสติคະແນນ Grader วิชา COM PROG ปี 2566

การสอบ	คະແນນເຈລ້ຍ		เตົມ
	ເກວມ 1	ເກວມ 2	
Grader 01	216.21	220.85	400
Grader 02	179.66	99.38	400
Grader 03	287.56	243.51	400
รวม	484.41	496.71	1200

# เกี่ยบสติคะแนน



## เกี่ยบสติคะแนน Midterm วิชา COM PROG ปี 2566

Midterm	เทอม 1	เทอม 2	เต็ม
Average	32.23	36.13	60
Maximum	57.00	54.00	60
Minimum	0.00	0.00	60

GET READY  
**FRESHY**



# Q&A

## תאמה טובים