

วัตถุประสงค์ เพื่อให้รู้จักโครงสร้างภาษาซี และการ compile และ ศึกษาการใช้ editor vi

### กิจกรรม

#### 1. ศึกษาการใช้ vi (มี 3 โหมด) ที่ prompt พิมพ์ vi

##### 1.1 command mode ใช้คีย์ i

1.1.1 เลื่อน cursor ไปบรรทัดบน ใช้คีย์ k

1.1.2 เลื่อน cursor ไปบรรทัดล่าง ใช้คีย์ j

1.1.3 เลื่อน cursor ไปทางขวา ใช้คีย์ L

1.1.4 เลื่อน cursor ไปทางซ้าย ใช้คีย์ h

1.1.5 ลบตัวอักษร ใช้คีย์ x

1.1.6 ลบบรรทัด ใช้คีย์ dd

1.1.7 แทรกบรรทัดข้างล่าง ใช้คีย์ c

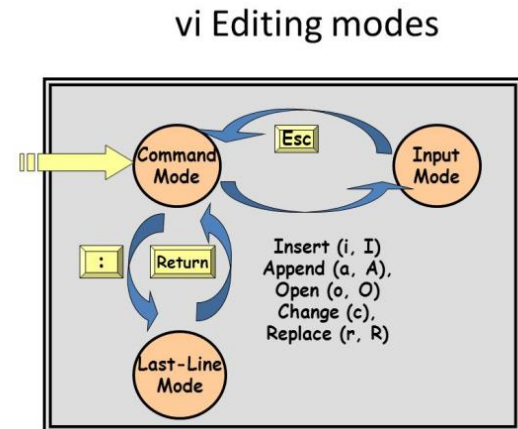
1.1.8 แทรกบรรทัดข้างบน ใช้คีย์ O (shift+o) \_  
(ข้อนี้ตั้งใจเฉลย)

1.1.9 คัดลอกทั้งบรรทัดใช้คำสั่ง yy

1.1.10 วางบรรทัดทั้งหมดที่คัดลอกไว้ ใช้คำสั่ง p

1.1.11 ยกเลิกคำสั่งล่าสุด (Undo) ใช้คำสั่ง C

##### 1.2 insert mode



1.2.1 ใช้คีย์ I (เข้า insert mode และ)

แทรกอักขรทางซ้าย

1.2.2 ใช้คีย์ A (เข้า insert mode และ)

แทรกอักขรทางขวา

1.2.3 ใช้คีย์ \_A (shift+a) (เข้า insert mode และ)

แทรกอักขรท้ายบรรทัด

##### 1.3 last line mode ใช้คีย์ e (จากโหมดคำสั่ง)

1.3.1 คำสั่ง save ใช้คีย์ w

1.3.2 คำสั่ง quit โปรแกรม ใช้คีย์ :q!

1.3.3 คำสั่งออกแบบ auto save wq, zz

### กิจกรรมที่ 2

#### 2.1 ไฟล์ lab3q2.c คำสั่งการรับค่า และแสดงผล

```

1  #include <stdio.h>
2  int main() {
3      int a,b,c;
4      printf("enter a number : ");
5      scanf("%d",&a);
6      printf("enter two number : ");
7      scanf("%d %d",&b, &c);
8      printf("summation = %d\n",a+b+c);
9      return 0;
10 }
```

#include เท่ากับ import

### กิจกรรมที่ 3

ศึกษาการใช้ github (เทอมนี้จะทดลองให้ส่งงานทาง github)

### กิจกรรมที่ 4

4.1 (cp lab3q2.c lab3q4\_xxyyy.c) ใช้ vi เขียนโปรแกรมเขียนโปรแกรมตามเงื่อนไขต่อไปนี้

4.1 ให้ผู้ใช้ป้อนเลข(จำนวนเต็มบวก)

4.2 โดยโปรแกรมจะจบการทำงานเมื่อผู้ใช้ป้อนข้อมูลเลข 0 หรือ ลบ

4.3 (ก่อนจบการทำงาน)แสดง ผลรวม และหาค่าเฉลี่ย ของตัวเลขที่ผู้ใช้ป้อน

### กิจกรรมที่ 5

5.1. ศึกษาการใช้ argc และ argv[]

The screenshot shows a C program named 'args.c' and its execution output. The program calculates the sum of command-line arguments. The output shows the program being compiled and run with different arguments, showing the values of argc and argv.

```

args.c
1 #include <stdlib.h>
2 #include <stdio.h>
3
4 int main(int argc, char* argv[])
5 {
6     int i, sum = 0;
7     printf("argc = %d\n", argc);
8     printf("Let's see what is in argv[]\n");
9     if(argc > 1)
10     {
11         for(i = 1; i < argc; i++)
12         {
13             printf("argv[%d] = %s\n", i, argv[i]);
14             sum += atoi(argv[i]);
15         }
16     }
17     return 0;
18 }

PaulProgramming: gcc args.c -o add
PaulProgramming: ./add
argc = 1
Let's see what is in argv[]
argv[0] = ./add
PaulProgramming: ./add 4 6 8
argc = 4
Let's see what is in argv[]
argv[0] = ./add
argv[1] = 4
argv[2] = 6
argv[3] = 8
PaulProgramming:
    
```

<https://i.ytimg.com/vi/aP1ijjeZc24/maxresdefault.jpg>

สังเกตจากผลการรัน 2 ครั้ง (console บรรทัดที่ 2 ./add และ บรรทัดที่ 6 ./add 4 6 8)

String args รับ parameter จากการสั่ง add ให้ทำงาน มาเก็บไว้ ส่วน atoi คือฟังก์ชันสำหรับแปลง String เป็น int (a หมายถึงอะไรที่ไม่ใช่ i (int) เป็น int atoi หมายถึงแปลงตัวเลขที่เป็นอักขระให้เป็น int)

### กิจกรรมที่ 6

ศึกษาโปรแกรมการคูณเมตริกซ์

#### คำสั่ง

ส่ง กิจกรรมที่ 1 (pdf) และ 4 (github URL)

กำหนดส่ง (TBA)

|                      |             |
|----------------------|-------------|
| (โปรดบันทึก)         |             |
| 1.1 _____            |             |
| 1.1.1 _____          | 1.2.1 _____ |
| 1.1.2 _____          | 1.2.2 _____ |
| 1.1.3 _____          | 1.2.3 _____ |
| 1.1.4 _____          |             |
| 1.1.5 _____          | 1.3 _____   |
| 1.1.6 _____          | 1.3.1 _____ |
| 1.1.7 _____          | 1.3.2 _____ |
| 1.1.8 _____ (วงเล็บ) |             |
| 1.1.9 _____          |             |
| 1.1.10 _____         |             |
| 1.1.11 _____         |             |