

ใบงานการทดลองที่ 3
เรื่อง คำสั่งพื้นฐานของภาษาซี โครงสร้าง และมาตรฐานการเขียนโค้ด

1. จุดประสงค์ทั่วไป

- 2.1. รู้และเข้าใจหลักภาษาที่ใช้ในการพัฒนาโปรแกรมคอมพิวเตอร์
- 2.2. รู้และเข้าใจมาตรฐานในการเขียนโค้ดโปรแกรมแบบ Codex

2. จุดประสงค์เชิงพฤติกรรม

- 2.1.1. บอกและอธิบายลักษณะของภาษาซี
- 2.1.3. บอกและอธิบายโครงสร้างของภาษาซี
- 2.1.4. บอกคำสั่งที่ใช้ในการการรับค่าและการแสดงผล
- 2.1.5. ฝึกหัดและทดลองใช้คำสั่งรับค่าและแสดงผล
- 2.2.1. บอกและอธิบายการจัดโครงสร้างของงาน
- 2.2.2. บอกและอธิบายการจัดโครงสร้างของโค้ดโปรแกรม
- 2.2.3. ฝึกหัดและทดลองใช้จัดโครงสร้างของโค้ดโปรแกรม
- 2.2.4. ออกแบบแนวทางการใช้จัดโครงสร้างของโค้ดโปรแกรมเพื่อให้ทำงานได้มีประสิทธิภาพมากขึ้น
- 2.2.5. แนะนำแนวทางการใช้จัดโครงสร้างของโค้ดโปรแกรมอย่างเป็นระบบ
- 2.2.6. บอกและอธิบายการตั้งชื่อตัวแปรให้สอดคล้องกับโครงสร้างหลัก
- 2.2.7. ฝึกหัดและทดลองใช้การตั้งชื่อตัวแปรให้สอดคล้องกับโครงสร้างหลัก
- 2.2.8. บอกและอธิบายการนิยามความหมายของตัวแปรและฟังก์ชัน
- 2.2.9. ฝึกหัดและทดลองใช้การนิยามความหมายของตัวแปรและฟังก์ชัน
- 2.2.10. บอกและอธิบายการเว้นวรรคตอน
- 2.2.11. ฝึกหัดและทดลองใช้การเว้นวรรคตอน
- 2.2.12. บอกและอธิบายการใช้งานเครื่องหมายวงเล็บ
- 2.2.13. ฝึกหัดและทดลองใช้การใช้งานเครื่องหมายวงเล็บ

3. เครื่องมือและอุปกรณ์

เครื่องคอมพิวเตอร์ 1 เครื่องที่ติดตั้งโปรแกรม Dev-C

4. ทฤษฎีการทดลอง

- 4.1. จงบอกและอธิบายที่มาและลักษณะพิเศษของ “ภาษาซี”

เป็นภาษาซีใช้สำหรับพัฒนาระบบ โปรแกรมคอมพิวเตอร์ที่มีขนาด
มีต้นกำเนิดในตระกูลภาษาซี ซึ่งใช้โครงสร้าง การออกแบบให้มี
ลักษณะเป็นภาษาที่ทำงานร่วมกับตัวแปร ประสิทธิภาพ คำสั่งใช้มีไม่
กำหนดโครงสร้างของข้อมูลในรูปแบบต่างๆ ภาษาซีถูกพัฒนา
ออกแบบเป็นรูปแบบ C++ มีความสามารถในการจัดการในภาษา
หรือวัตถุ นอกจากนี้ ภาษาซี ยังถูกใช้เพื่อเป็นต้นแบบของ ภาษา
Java, PHP, Python, javascript.

4.2. จงบอกและอธิบายโครงสร้างของภาษาซีโดยทั่วไป
mr. ฝึกโปรแกรมคอมพิวเตอร์ มักจะจำเป็นจะต้องเขียนในภาษาซี
ตามหลักไวยากรณ์ (syntax) ของภาษานั้น ซึ่งในภาษาซีจะมี
ภาษาที่ไว้เป็นพื้นฐานสำหรับเขียนโปรแกรม
เช่น printf ใช้แสดงผลการเขียน

4.3. จงบอกและอธิบายลักษณะการใช้งานคำสั่ง printf พร้อมยกตัวอย่างประกอบ

printf เป็นmr.แสดง โปรแกรมพิมพ์บนหน้าจอ.
เช่น printf ("Hello world");
อยู่ในคำสั่งหลัก ของstdio <stdio.h>

4.4. จงบอกและอธิบายลักษณะการใช้งานคำสั่ง scanf พร้อมยกตัวอย่างประกอบ

scanf เป็นmr.รับข้อมูลเพื่อ รับค่าส่งของของ
ตัวแทนที่ (Place holder) และ ส่วนของตัวแปร (variable)
scanf ("ตัวแทน 1, ตัวแทน 2", ตัวแปร 1, ตัวแปร 2);
scanf ("%d, %f", &a, &b);

4.5. จงบอกและอธิบายการจัดโครงสร้างของโค้ดโปรแกรมตามมาตรฐาน Codex พร้อมยกตัวอย่างประกอบ

เป็นมาตรฐานใน Wordpress CMS ซึ่งเป็นโปรแกรมสร้าง
เว็บไซต์ (แบบสำเร็จรูป) ที่พัฒนาโดยใช้ภาษา PHP หรือสคริปต์ใช้
งานที่ระดม ช่างต่างราย เพื่อเปิดโอกาสให้นักพัฒนา รวบรวมทุกๆ
ทั่วโลกมาช่วยสร้างและพัฒนาการทำงานร่วมกันกับระบบหลักได้

4.6. จงบอกและอธิบายการตั้งชื่อตัวแปรตามมาตรฐาน Codex พร้อมยกตัวอย่างประกอบ

mr.ตั้งชื่อโครงงาน จะต้องตั้งด้วยชื่อภาษาอังกฤษ (ตัวพิมพ์เล็ก)
mr.ตั้งชื่อแม่ข่ายหลัก, แม่ข่ายย่อย, ชื่อโปรแกรมย่อย, ชื่อตัวแปร
ต้องตั้งให้ตรงกับชื่อโครงงาน แล้วจึงใช้ชื่อตัวแปรประกอบ
เช่น calculator for student html, calculator for student loop. เป็นต้น.

4.7. จงบอกและอธิบายการนิยามตัวแปรและฟังก์ชันตามมาตรฐาน Codex พร้อมยกตัวอย่างประกอบ

การนิยามตัวแปร และ function จะต้องตั้งให้สอดคล้องกับหัวข้อ
 เช่น `int m, n;` = ตัวแปร m, n.
`void main` = ฟังก์ชัน main

4.8. จงบอกและอธิบายการเว้นวรรคตามมาตรฐาน Codex พร้อมยกตัวอย่างประกอบ

ทุกในคำสั่งต้องต่างกัน จะต้องถูกจัดเรียงตัวกัน เว้นวรรค
 เช่น `int m, n;` `void main`
`int m, n;` `void main`
 (2), `int m, n;` `void main`

4.9. จงบอกและอธิบายการใช้งานเครื่องหมายวงเล็บตามมาตรฐาน Codex พร้อมยกตัวอย่างประกอบ

เมื่อพบเครื่องหมายวงเล็บปีกกา เปิด ({) จะต้องบอกให้เห็น
 โครงสร้างของชุดโปรแกรม เมื่อปิดโปรแกรม หรือพบปีกกาปิด }
 ก็จะถือว่าสิ้นสุด "ช่องหน้า" เมื่อใช้ปีกกาปิดโปรแกรมอยู่ภายใน
 โครงสร้างที่จบแล้ว จะกลับเข้าสู่โครงสร้างเดิม เมื่อพบเครื่องหมายวงเล็บ (}

5. ลำดับขั้นการปฏิบัติงาน

5.1. การรับและการแสดงผล

5.1.1. จงเขียนโปรแกรมเพื่อรับตัวเลขจำนวนเต็ม ตัวเลขทศนิยม และตัวอักษร เพื่อแสดงผลบนหน้าจอคอมพิวเตอร์

Test case 1	Test Case 2
Input Please enter value : 5 12.23945 F	Input Please enter value : 9 13.1 M
Output 5 12.24 F	Output 9 13.10 M

5.1.2. บันทึกโค้ดโปรแกรมดังกล่าว

ผังงาน	โค้ดโปรแกรม
<pre> graph TD Start([start]) --> N[int N] N --> sum[float sum] sum --> x[chr x] x --> Input[/Please enter value/] Input --> Output[/N, sum, x/] Output --> Print[/N, sum, X/] Print --> Stop([stop]) </pre>	<pre> #include <stdio.h> int main () { int N; float sum; char x; Printf("Please enter value : "); scanf("%d %f %c", &N, &sum, &x); Printf("%d %5.2f %c", N, sum, x); return 0; } // end </pre>

5.2. มาตรฐานการเขียนโค้ดโปรแกรมตามมาตรฐาน Codex

5.2.1. จากโค้ดโปรแกรมต่อไปนี้ จงจัดระเบียบให้อยู่ภายใต้มาตรฐานการเขียนโปรแกรมแบบ Codex พร้อมทั้งแก้ไขอาการผิดปกติ (Bug) เพื่อให้โปรแกรมสามารถทำงานได้ตามปกติ

โค้ดโปรแกรม
<pre> #include <stdio.h> int main() { int i=0; int N=7; printf("-- Show 'Hello World' to Screen %d line(s) \n", N); for(i=0; i<N; i++) { printf("[%3d] Hello World \n", i);} char star="*" int j = 0 ; N = 8 ; i = 0 ; j = 0 ; printf("\n"); while(i < N) { j = 0 ; while(j < N) { if(i < j) printf("%c ", star); j++ ; } printf("%s", "\n" ; i += 1 ; } return 0 ; } </pre>

5.2.2. บันทึกผลโค้ดโปรแกรมที่ถูกจัดระเบียบแล้วลงตารางดังต่อไปนี้

โค้ดโปรแกรม

```
#include <stdio.h>
int main () {
    int i = 0;
    int N = 7;
    printf("--) show 'Hello world' to screen x-d lines\n", N);
    for (i = 0; i < N; i++) {
        printf("[x-d] Hello world\n", i);
    }
    char star = '*';
    int b = 8;
    int c = 0;
    int d = 0;
    printf("\n");
    while (c < b) {
        d = 0;
        while (d < b) {
            if (c < d)
                printf("%c", star);
            d++;
        }
        printf("%s ", "\n");
        c++;
    }
    return 0;
} // end main.
```

5.2.3. บันทึกผลลัพธ์การทำงานของโค้ดโปรแกรมนี้

ผลลัพธ์การทำงานของโปรแกรม

```
[0] Hello world
[1] Hello world
[2] Hello world
[3] Hello world
[4] Hello world
[5] Hello world
[6] Hello world
[7] Hello world
```



```

      X
* * * * *
* * * * *
* * * * *
* * * * *
* * * * *
* * * * *
* * * * *
* * * * *

```

6. สรุปผลการปฏิบัติงาน

Printf = การแสดงผลออกหน้าจอ.

scanf = การรับข้อมูลทางจอ แล้วนำมาแสดงผล.

การมาบรรณ การเขียนของ codex การมีปีกกา การเว้นวรรคต่างๆ
เมื่อแก้ข้อ code

7. คำถามทางการทดลอง

7.1. จงบอกจุดผิดพลาดที่ทำให้มักเกิดสิ่งผิดปกติภายในโค้ดโปรแกรมบ่อยที่สุด พร้อมแนวทางการแก้ไขปัญหาที่เกิดขึ้น

จุดผิดพลาดบ่อยที่สุด คือ การกำหนดตัวแปร
การลืมเครื่องหมาย ; หรือการใช้ปีกกาไม่ครบ
แนวทางการแก้ไข นั้นตรวจสอบให้ดีก่อนพิมพ์ แล้วจึง
เขียนโปรแกรมก่อน

7.2. จงบอกประโยชน์ของมาตรฐานการเขียนโค้ดโปรแกรมแบบ Codex พร้อมยกตัวอย่างประกอบ

ประโยชน์ - มีรูปแบบที่ชัดเจน - มีระเบียบ - แก้ไขง่าย

เช่น การใช้ปีกกาบ่งบอกคู่กับ ฟังก์ชัน หรือส่วนหัว

```
int main() {
```

```
→ printf("P");
```

```
}
```

เป็นต้น.