

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"

มีการกำหนดสัณฐาน และทางเรขาคณิตของตัวเอง

4.2. จงบอกและอธิบายโครงสร้างของภาษาซีโดยทั่วไป

ประกอบด้วย 3 ส่วนหลักๆ ได้แก่

1. ส่วนหัว ส่วนสำหรับตัวชี้แจง

2. ประกาศตัวแปร กำหนดชนิดของตัวแปรที่ใช้

3. ส่วนหลัก

4.3. จงบอกและอธิบายลักษณะการใช้งานคำสั่ง printf พร้อมยกตัวอย่างประกอบ

เป็นคำสั่งที่นำข้อมูลไปแสดงผลลัพธ์ในหน้าจอคอมพิวเตอร์

โดยเป็นพารามิเตอร์, ทิศทาง, ตัวชี้แจง

4.4. จงบอกและอธิบายลักษณะการใช้งานคำสั่ง scanf พร้อมยกตัวอย่างประกอบ

เป็นฟังก์ชันนำค่าที่ป้อนจากผู้ใช้มาเก็บไว้ในตัวแปรที่กำหนด

ทศนิยม, ว่าง

4.5. จงบอกและอธิบายการจัดโครงสร้างของโค้ดโปรแกรมตามมาตรฐาน Codex พร้อมยกตัวอย่างประกอบ

เพื่อให้สามารถสื่อสารกับนักพัฒนาคนอื่นได้ การจัดทำเอกสาร

ประกอบไปด้วย การตั้งชื่อตัวแปร การตั้งชื่อฟังก์ชัน การตั้งชื่อไฟล์

การตั้งชื่อในวงเล็บ (เพิ่มเติม)

4.6. จงบอกและอธิบายการตั้งชื่อตัวแปรตามมาตรฐาน Codex พร้อมยกตัวอย่างประกอบ

การตั้งชื่อตัวแปรในโปรแกรม เช่น loop 2.2 - loop นั้น

4.7. จงบอกและอธิบายการนิยามตัวแปรและฟังก์ชันตามมาตรฐาน Codex พร้อมยกตัวอย่างประกอบ

ตัวแปรคือค่าที่อยู่ในหน่วยความจำ โดยจะเปลี่ยนค่าได้ตามความต้องการ
ฟังก์ชันคือ ทำให้งาน ที่มีการ กำหนดค่า เข้ามา และให้ค่าออกมา
ค่าที่ออกมาถูกต้อง

4.8. จงบอกและอธิบายการวนซ้ำตามมาตรฐาน Codex พร้อมยกตัวอย่างประกอบ
หรือได้เน้นโครงร่าง ทวไปมา และวิธีแปรค่าของตัวแปรให้ค่าที่
เช่น เริ่มที่ค่า 1 แล้วเพิ่มค่าขึ้นเรื่อยๆ จนถึงค่าที่กำหนด

4.9. จงบอกและอธิบายการใช้งานเครื่องหมายวงเล็บตามมาตรฐาน Codex พร้อมยกตัวอย่างประกอบ

เป็นการใช้เพื่อกำหนดเงื่อนไข หรือเน้นค่าที่ต้องการได้รวดเร็ว ปกติ เช่น
เช่น หาค่าของค่า

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([เริ่ม]) --> Input[/รับค่า Input จาก keyboard หรือ รับค่า int, float, char/] Input --> Process[/จัดเก็บค่า ป้อนในหน่วยความจำ/] Process --> Process2[/นำค่าที่ป้อนมา แสดงในจอ 1 วินาที/] Process2 --> Process3[/นำข้อมูลมา แสดงผล/] Process3 --> End([สิ้นสุด]) </pre>	<pre> #include <stdio.h> int main() { int a; float b; char c; printf("Please enter values"); scanf("%d %f %c", &a, &b, &c); printf("r.d %d %f %c", a, b, c); return 0; } </pre>

5.2. มาตรฐานการเขียนโค้ดโปรแกรมตามมาตรฐาน Codex

5.2.1. จากโค้ดโปรแกรมต่อไปนี้ จงจัดระเบียบให้อยู่ภายใต้มาตรฐานการเขียนโปรแกรมแบบ Codex พร้อมทั้งแก้ไขข้อบกพร่อง (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("--I show 'Hello World' to screen %d line(s)\n", N);
    for (i = 0; i < N; i++) {
        printf("[%d] 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);
        }
        printf("%s", "\n");
        i = i + 1;
    }
    return 0;
}
```


5.2.3 บันทึกผลลัพธ์การทำงานของโค้ดโปรแกรมนี้

ผลลัพธ์การทำงานของโปรแกรม

พิมพ์ Hello World 7 บรรทัด

```

*
*
*
*
*
*
*

```

6. สรุปผลการปฏิบัติงาน

ในการเขียนโปรแกรมต้องมีการทดสอบโปรแกรมให้แน่ใจว่าโปรแกรมทำงานได้ตามที่ต้องการ

7. คำถามทางการทดลอง

7.1. จงบอกจุดผิดพลาดที่ทำให้เกิดสิ่งผิดปกติภายในโค้ดโปรแกรมบ้อยที่สุด พร้อมแนวทางการแก้ไขปัญหที่เกิดขึ้น

จุดผิดพลาดในการเขียนโปรแกรม เช่น การใส่เครื่องหมายผิด หรือการใส่เครื่องหมายผิด...

7.2. จงบอกประโยชน์ของมาตรฐานการเขียนโค้ดโปรแกรมแบบ Codex พร้อมยกตัวอย่างประกอบ

การตั้งชื่อตัวแปรให้มีความหมายและสอดคล้องกับสิ่งที่ทำอยู่