

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"

..... เป็นภาษาที่ออกแบบมาสำหรับคอมพิวเตอร์ ใช้ในการเขียน
โปรแกรม ส่วนใหญ่ มีลักษณะโปรแกรมเชิงโครงสร้าง ถูกคิดค้นขึ้น
19 61- 19 63 โดย ดอนัลด์ ดี. ริทซ์ ออกแบบมาเพื่อให้นักวิจัย
การวิจัย ใช้ตาม สถาบันวิจัย การศึกษา ในทางวิศวกรรม

4.2. จงบอกและอธิบายโครงสร้างของภาษาซีโดยทั่วไป

ภาษาซีประกอบด้วย

- ส่วนหัว หรือ header

- โปรแกรมหลัก main

- ส่วน output

- ส่วนตัวแปร และ constant

1. Include 2. Function

3. Prototype

4. Global variable

5. Structure

6. Main function

4.3. จงบอกและอธิบายลักษณะการใช้งานคำสั่ง printf พร้อมยกตัวอย่างประกอบ

คำสั่ง printf ใช้สำหรับส่งข้อมูลไปยังหน้าจอ ("ข้อความที่ต้องการ")

ตัวอย่างการใช้งาน

printf("ส่วนหัว");

4.4. จงบอกและอธิบายลักษณะการใช้งานคำสั่ง scanf พร้อมยกตัวอย่างประกอบ

คำสั่ง scanf ใช้สำหรับรับข้อมูลจากแป้นพิมพ์

ของโปรแกรมเมอร์ที่ป้อนเข้ามาในโปรแกรม และจะเก็บค่าที่ป้อนไว้ที่ตำแหน่งที่เรากำหนด (Place holder) เพื่อใช้ในการประมวลผลต่อไป

1. scanf("ข้อความ", &ตัวแปร);

4.5. จงบอกและอธิบายการจัดโครงสร้างของโค้ดโปรแกรมตามมาตรฐาน Codex พร้อมยกตัวอย่างประกอบ

การจัดโครงสร้างของโปรแกรม

การจัดโครงสร้างของโปรแกรมให้มีความเป็นระเบียบและง่ายต่อการอ่านและแก้ไข

calculator_for_student การคำนวณเกรดเฉลี่ย

calculator.html การคำนวณเกรดเฉลี่ย

4.6. จงบอกและอธิบายการตั้งชื่อตัวแปรตามมาตรฐาน Codex พร้อมยกตัวอย่างประกอบ

การตั้งชื่อตัวแปรให้มีความเป็นระเบียบและง่ายต่อการอ่านและแก้ไข

calculator_for_student.html

ประกอบ

calculator for student

ส่วนที่ ๑ การจัดทำบัญชีและงบการเงิน

7. What is the purpose of the following?

61

ประกอบ

၁။ အောက်ဖော်ပြပါ အချက်များကို စစ်ဆေးပြီးနောက်
 သေချာ ဂရုစိုက်မှု (၆) နည်းပါးသော အချက်များကို
 ဖော်ပြပါ။

Ex (3)

5. ลำดับชั้นการปฏิบัติงาน

5.1. การรับและการแสดงผล

บนหน้าจอคอมพิวเตอร์

<u>Test case 1</u>	<u>Test Case 2</u>
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([เริ่มงาน]) --> Read[รับค่าจากแป้น] Read --> Calc[คำนวณ] Calc --> Output[แสดงผล] Output --> End([สิ้นสุด]) </pre>	<pre> #include <stdio.h> #include <conio.h> int main() { char c; float ff; int i; printf("input 1 :"); scanf("%d", &i); fflush(stdin); printf("input 2 :"); scanf("%f", &ff); fflush(stdin); printf("input 3 :"); scanf("%c", &c); fflush(stdin); printf("%d, %.2f %c", i, ff, c); return 0; } </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>

โค้ดโปรแกรม

```

#include <stdio.h>

int main() {
    int i = 0;
    int N = 7;

    printf("-- | show 'Hello World' to screen %d lines\n", N);
    for (i = 0; i < N; i++) {
        printf("[%d] Hello World\n", i);
    } // end function for
    char star = '*';
    int j = 0;

    N = 4;
    i = 0;
    j = 0;
    printf("\n");
    while (i < N) {
        j = 0;
        while (j < N) {
            if (1 < j)
                printf(" ", star);
            j++;
        } // end function j < N
        printf("\n");
        i++;
    } // end function i < N
    return 0;
} // end function main

```

5.2.3. บันทึกผลลัพธ์การทำงานของโค้ดโปรแกรมนี้

ผลลัพธ์การทำงานของโปรแกรม

```

* * * * *
* * * * *
* * * * *
* * * * *
* * * * *
* * * * *
* * * * *
* * * * *
* * * * *

```

6. สรุปผลการปฏิบัติงาน

จากการทำงาน พบว่า รหัสโปรแกรม 01000001 โปรแกรม 01000001
 01000001 โปรแกรม 01000001 โปรแกรม 01000001

7. คำถามทางการทดลอง

7.1. จงบอกจุดผิดพลาดที่ทำให้มักเกิดสิ่งผิดปกติภายในโค้ดโปรแกรมบ่อยที่สุด พร้อมแนวทางการแก้ไขปัญหาที่เกิดขึ้น

รหัส โปรแกรม 01000001 โปรแกรม 01000001
 โปรแกรม 01000001 โปรแกรม 01000001 โปรแกรม 01000001
 โปรแกรม 01000001 โปรแกรม 01000001 โปรแกรม 01000001

7.2. จงบอกประโยชน์ของมาตรฐานการเขียนโค้ดโปรแกรมแบบ Codex พร้อมยกตัวอย่างประกอบ

จาก รหัส 01000001 โปรแกรม 01000001 โปรแกรม 01000001
 โปรแกรม 01000001 โปรแกรม 01000001 โปรแกรม 01000001