

แผนการสอนปฏิบัติครั้งที่ 3

- ✚ **วัตถุประสงค์** เข้าใจความสัมพันธ์ระหว่างผังงานกับโปรแกรม/เขียนโปรแกรม/ดัดแปลงโปรแกรม
- ✚ **โจทย์ปัญหา** ให้นำผังงานโปรแกรมทายตัวเลขที่กำหนดให้ มาเขียนเป็นโปรแกรมภาษาซี
- ✚ **ผลลัพธ์การเรียนรู้**
 - นศ. สามารถแปลงผังงาน มาเป็นคำสั่งโปรแกรมภาษาซีได้
- ✚ **ขั้นตอน**
 - ทำความเข้าใจการเล่นเกมทายตัวเลข ตามเงื่อนไขที่กำหนด (สอดคล้องกับผังงาน)
 - สุ่มตั้งตัวเลขจำนวนเต็ม 1-100 ให้ทายได้ไม่เกิน 7 ครั้ง
 - ถ้าทายถูกภายใน 7 ครั้ง สรุปรว่าชนะ ถ้าไม่ถูก สรุปรว่าแพ้ และให้เฉลยตัวเลขที่ถูกต้อง
 - ทำความเข้าใจคำสั่งที่สอดคล้องกับผังงาน (ใช้ผังงานที่กำหนดให้)
 - `srand(time(NULL)); x = rand() %100 +1 ; //ตัวอย่างคำสั่งสุ่มค่า x ให้มีค่า 1..100 ต้อง #include<time.h>`
 - แปลงผังงานเป็นโปรแกรม (ต้องเขียนโปรแกรมให้สอดคล้องกับผังงานที่ออกแบบไว้)
- ✚ **งานที่ต้องส่งให้ดูในคาบ** หน้าจอแสดงผลที่ทดสอบ
 - การตั้งคำถามจะต้องมีการแสดงจำนวนครั้งที่กำลังทายอยู่ (1-7)
 - กรณีป้อนข้อมูลเป็นตัวเลขถูกต้อง 3 กรณี คือ
 - 1. กรณีทายถูกก่อน 7 ครั้ง 2. กรณีทายถูกในครั้งที่ 7 และ 3. กรณีทายผิดครั้งที่ 7
 - กรณีป้อนข้อมูลผิดหรือเกิน เช่น [120], [5.0], [5 6 7], [5x], [5 x] สรุปรผล/ความผิดพลาดที่ทำให้โปรแกรมทำงานไม่ตรงกับที่คาดไว้
- ✚ **รายงานที่ต้องส่งตามหลัง** (ต้องสรุปผลการประเมินตนเองและปัญหาในการทำ Assignment ทุกครั้ง)
 - ดัดแปลงโปรแกรม(User Interface) ให้มีตั้งคำถาม และสรุปคำตอบที่อ่านเข้าใจง่าย
 - รายงานอธิบายการทำงานของโปรแกรมให้สอดคล้องกับผังงานที่เขียน และยกตัวอย่างหน้าจอแสดงผลของทั้ง 4 กรณี
 - โค้ดภาษา C ของโปรแกรมที่สมบูรณ์แล้ว

ผังงานเกมส่ายตัวเลข

จองตัวแปร
int x, y, count ;

```
srand(time(NULL));  
x = rand() % 100 + 1;  
count = 0;
```

การวนกลับเพื่อทำซ้ำ

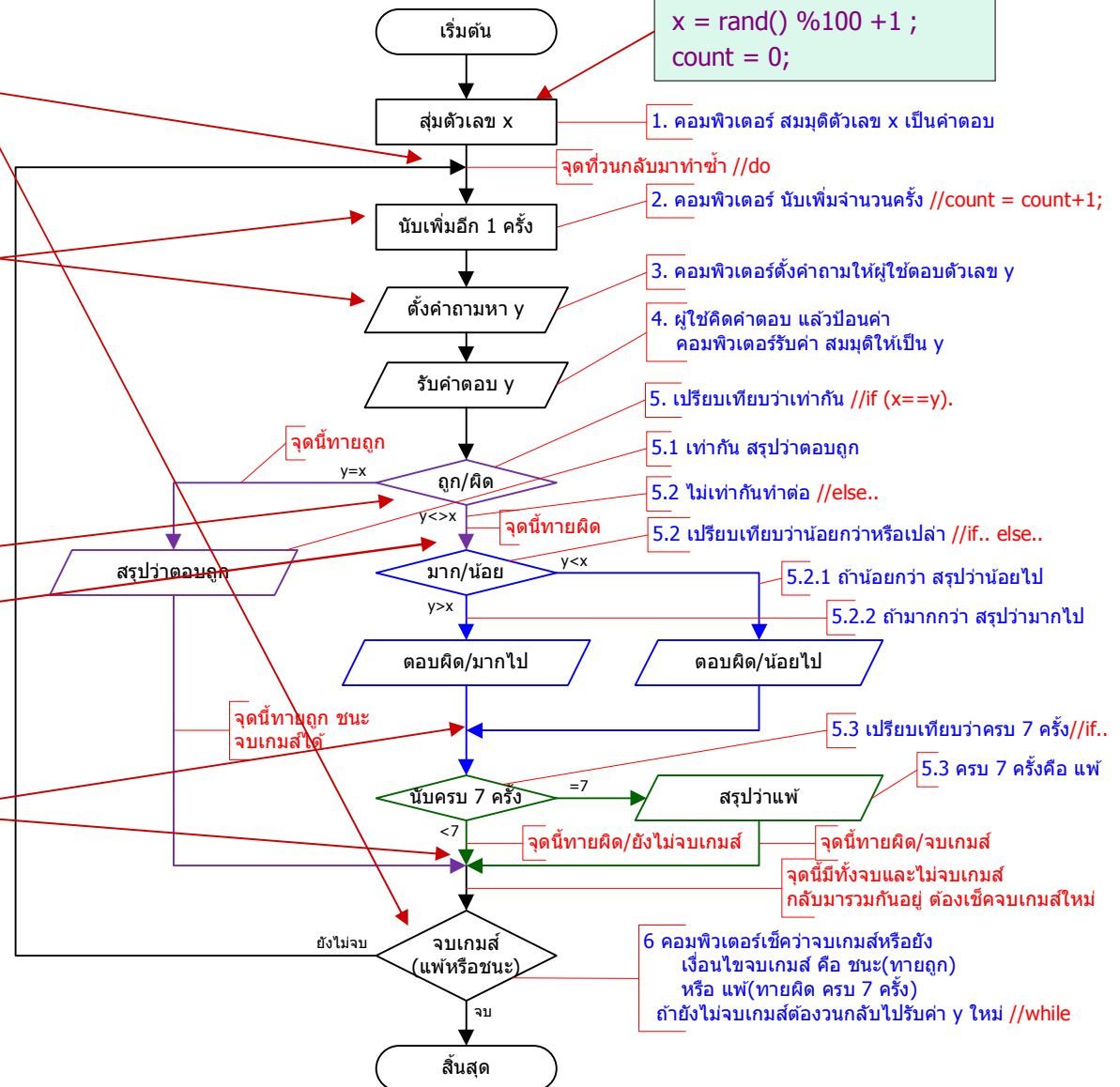
- ต้องมีจุดเริ่มต้น
- ต้องมีจุดตัดสินใจ

การรอรับคำตอบ
ต้องตั้งคำถามก่อน

การตัดสินใจ

- แยกเป็น 2 ทางเลือก
- ถ้าต้องการ 3 ทางเลือก ให้ตัดสินใจซ้อน 2 ครั้ง

ทางที่แยกออกจากการตัดสินใจ 2 ทาง ต้องกลับมารวมกัน



ลำดับวิธีคิดตามผังงานโปรแกรมทายตัวเลข

✚ ผู้ตั้งคำถาม คือ คอมพิวเตอร์ (Hardware) คนที่ 2 คือ ผู้เล่น(User) ตัวเกมส์ คือ ซอฟต์แวร์ (Software)

1. คอมพิวเตอร์ **สมมติตัวเลข** เป็นคำตอบ (คอมพิวเตอร์ต้องสุ่มตัวเลขให้ทาย)
2. คอมพิวเตอร์ เริ่มนับเพิ่มครั้งที่ทาย (คอมพิวเตอร์บวกจำนวนครั้งเพิ่ม 1)
3. คอมพิวเตอร์ **ตั้งคำถาม** ให้ผู้ใช้ตอบตัวเลข (คอมพิวเตอร์ต้องแสดงผลคำถาม)
4. ผู้เล่นคิดตัวเลขแล้วตอบ **โดยป้อนตัวเลขให้คอมพิวเตอร์** (คอมพิวเตอร์ต้องรับตัวเลขไปเก็บ)
5. คอมพิวเตอร์ **เปรียบเทียบ** ตัวเลขของผู้เล่น กับคำตอบ
 - 5.1 ถ้า **เท่ากัน** แสดงว่าตอบถูก **สรุปว่าชนะเพื่อจบเกมส์** (จบเกมส์ได้)
 - 5.2 ถ้า **ไม่เท่า** แสดงว่าตอบผิด ยังต้องคิดว่าทายมากไป หรือน้อยไป
 - 5.2.1 ถ้าตัวเลข **น้อยกว่า** คำตอบ **สรุปว่าทายน้อยไป**
 - 5.2.2 ถ้าตัวเลข **มากกว่า** คำตอบ **สรุปว่าทายมากไป**
 - 5.3 ถ้าผิดครบ **7 ครั้ง** ให้สรุปว่า **แพ้เพื่อจบเกมส์** (ถ้ายังไม่ครบไม่ต้องสรุป)
6. ถ้า **ยังไม่จบเกมส์** (ตอบผิดน้อยกว่า 7 ครั้ง) จะต้องวนกลับไปทำซ้ำขั้นตอนการเดิม คือนับเพิ่ม และตั้งคำถามใน step 2 จนถึงสรุปใน step 5
7. จบโปรแกรมได้ เมื่อจบเกมส์แล้ว (ผู้ใช้ตอบถูก หรือทายผิดครบ 7 ครั้ง)

เงื่อนไขเพิ่มเติม(บังคับน.ศ. CPE)

- ✚ ต้องเขียนโปรแกรม Assignment 3 ตามผังงานให้เสร็จก่อน (สามารถส่งตรวจก่อนในขั้นตอนนี้ได้)
- ✚ ดัดแปลงโปรแกรมเพื่อแก้ไขจุดบกพร่องและทดสอบให้เสร็จที่ละหัวข้อ ตามลำดับ
 1. เพิ่มการแนะนำตัวของโปรแกรมว่า เป็นโปรแกรมทายตัวเลข
 2. เพิ่มคำถาม เพื่อถามชื่อผู้เล่น และมีการทักทายผู้เล่น
 3. เปลี่ยนแปลงคำถาม คำตอบ ให้มีการระบุชื่อผู้เล่นด้วย
 4. โปรแกรมต้องสามารถป้องกันความผิดพลาดจากการป้อนค่าที่ไม่ต้องการ (Test case 4) ถ้าไม่ถูกต้องให้รอรับใหม่ **โปรแกรมต้องไม่หยุดทำงาน หรือนำค่าที่ผิดไปใช้** ให้ตัดสินใจเรื่องความผิดพลาดที่ยอมรับได้หรือไม่ได้เองว่าจะให้ทำงานหรือไม่เช่น [120], [5.0], [5 6 7], [5x], [5 x] พร้อมสรุปเงื่อนไขที่รับได้/ไม่ได้
 5. เมื่อเล่นจบ ให้โปรแกรม เพิ่มคำถามว่าจะเริ่มเล่นใหม่หรือไม่
 1. โปรแกรมรับเฉพาะปุ่ม [n] , [N] , [y] หรือ [Y] เท่านั้น ถ้ากดผิดปุ่มให้รอรับใหม่
 2. ถ้ากดปุ่ม [y] หรือ [Y] แสดงว่าเล่นต้องการเล่นซ้ำ ให้วนกลับไปสู่ตัวเลขเริ่มต้นใหม่ แต่ไม่ต้องถามชื่อผู้เล่นซ้ำอีก ถ้าไม่ใช่ให้จบโปรแกรมได้
- ✚ งานที่ส่งเข้า LEB2 จะเป็นรายงานและ code ของโปรแกรมที่ปรับปรุงแล้ว

```
char name[30];  
printf("Enter you name ");  
scanf("%s",name);  
printf("Hello %s",name);
```