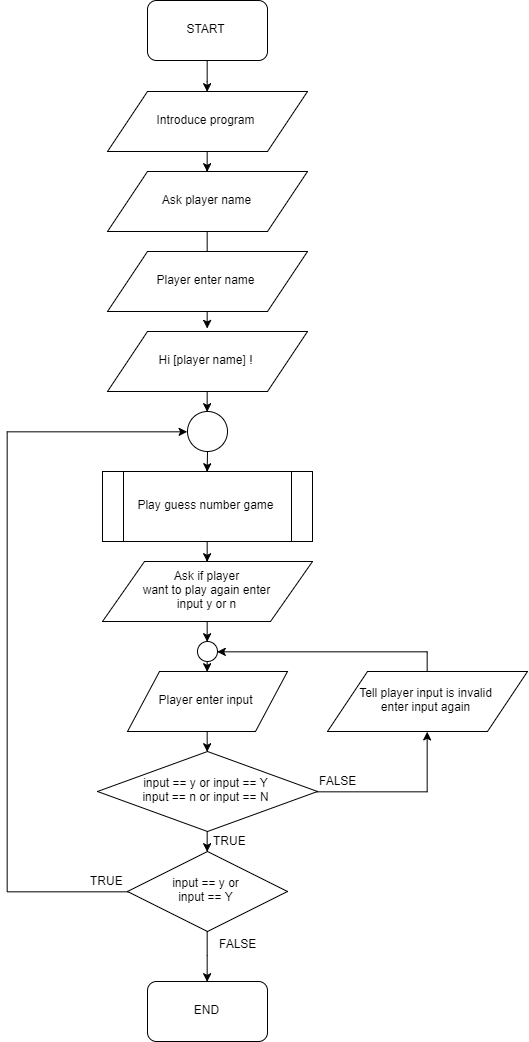
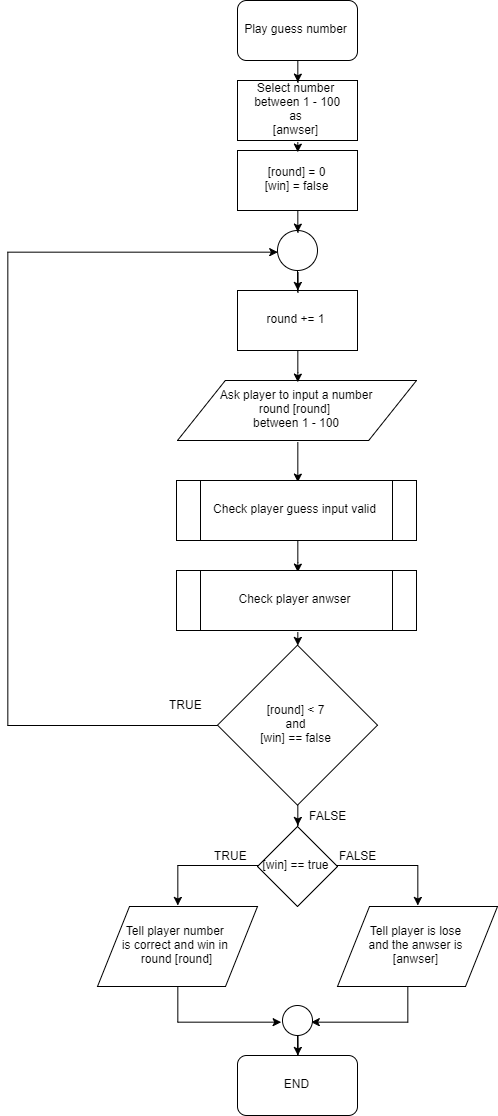
**Assignment 3 เกมทายตัวเลข**

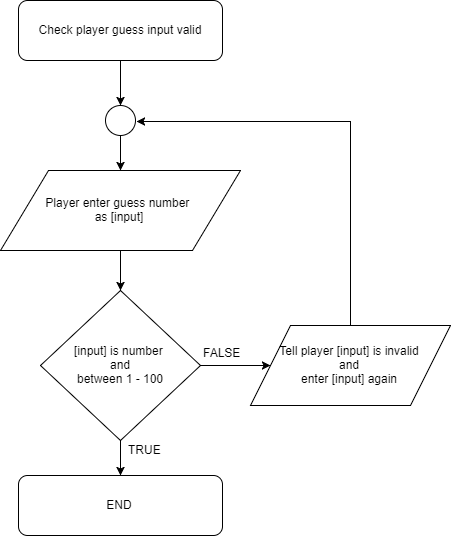
**Flowchart ส่วนหลัก**

****

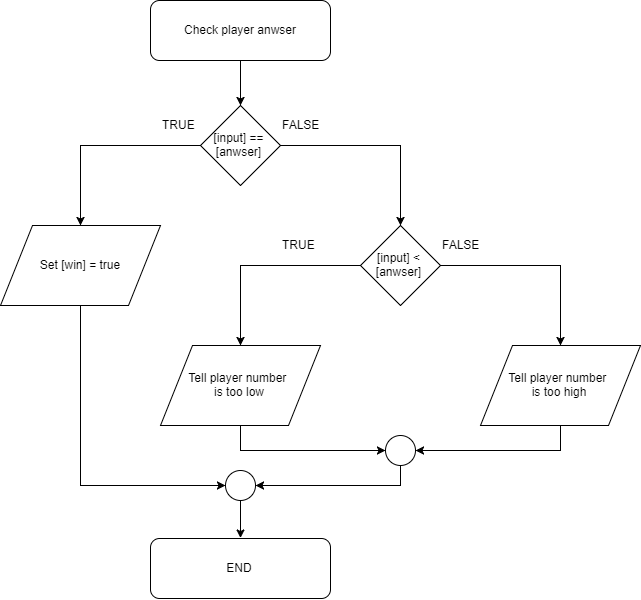
**Flowchart ส่วน Play guess number game**

****

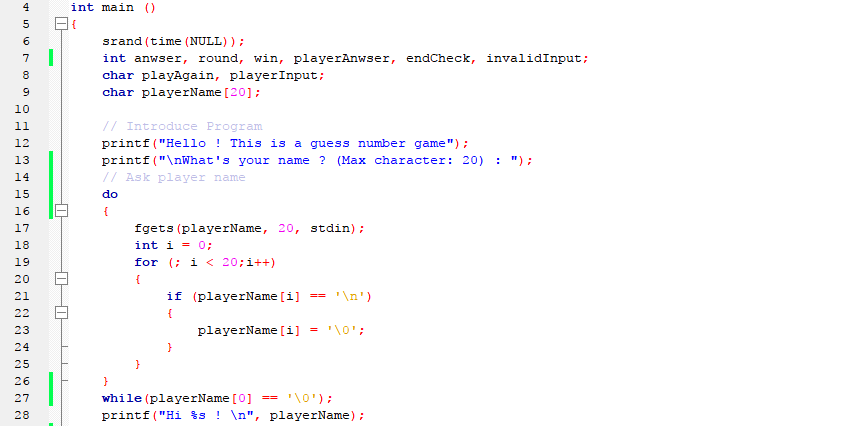
**Flowchart ส่วน Check player guess number input**

****

**Flowchart ส่วน Check player answer**

****

**อธิบายการทำงานของโปรแกรม**



**บรรทัดที่ 6 :** กำหนดค่า seed เพื่อใช้ในการสุ่มค่าตัวเลข

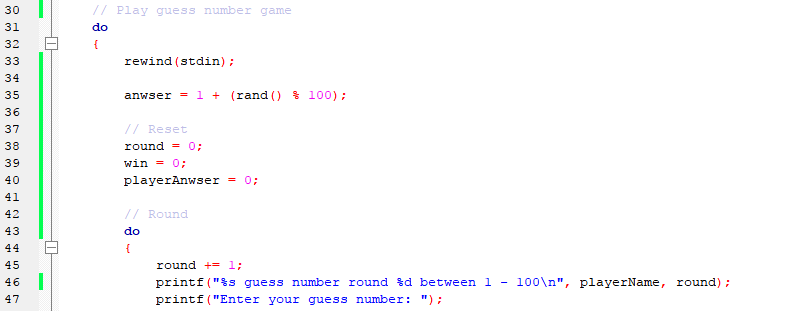
**บรรทัดที่ 7 – 9 :** ประกาศตัวแปรเตรียมนำไปใช้งาน

**บรรทัดที่ 12 :** โปรแกรมแนะนำตนเอง

**บรรทัดที่ 13 :** โปรแกรมถามชื่อผู้เล่น

**บรรทัดที่ 14 – 27 :** เก็บ input ชื่อผู้เล่น

**บรรทัดที่ 28 :** โปรแกรมทักทายผู้เล่น โดยกล่าวชื่อ



**บรรทัดที่ 31 :** เริ่มลูป do while สำหรับเล่นเกมซ้ำ

**บรรทัดที่ 33 :** ลบค่าที่ค้างอยู่ใน buffer ป้องกันค่าที่อยูใน buffer ทำให้โปรแกรมทำงานผิดพลาด จากการเล่นซ้ำ

**บรรทัดที่ 35 :** สุ่มเลือกตัวเลขระหว่าง 1 – 100

**บรรทัดที่ 38 - 40 :** รีเซ็ตค่าตัวแปรเพื่อเริ่มเล่นเกม

**บรรทัดที่ 43 :** เริ่มลูป do while สำหรับรอบการทายตัวเลข

**บรรทัดที่ 45** **:** นับรอบที่เล่นอีก 1 ครั้ง

**บรรทัดที่ 46 :** บอกผู้เล่นให้ทายเลขมาหนึ่งเลขระหว่าง 1 – 100 พร้อมเรียกชื่อผู้เล่นและบอกรอบปัจจุบัน

**บรรทัดที่ 47** **:** บอกให้ผู้เล่นกรอกเลขที่คิดไว้



**บรรทัดที่ 50 :** เริ่มลูป do while สำหรับตรวจสอบค่าที่ผู้เล่นกรอกมา

**บรรทัดที่ 52 – 55 :** กำหนดตัวแปรขึ้นมาเพื่อใช้เช็คว่า ค่าที่ผู้เล่นกรอกถูกต้องหรือไม่ และใช้เก็บค่าที่ผู้เล่นกรอกมา

**บรรทัดที่ 56 :** เริ่มลูป while ทำงานเมื่อค่า endCheck เท่ากับ 0 หรือก็คือยังไม่จบการตรวจสอบ

**บรรทัดที่ 58 :** อ่านค่าที่ผู้เล่นกรอกมาทีละตัวเก็บไว้ใน playerInput

**บรรทัดที่ 59 :** เช็คว่า playerInput มาเป็นตัวเลขหรือเป็นการขึ้นบรรทัดใหม่หรือไม่

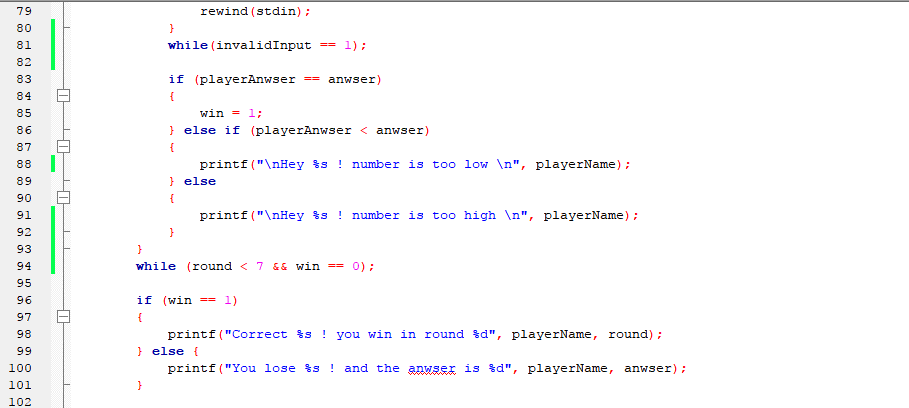
**บรรทัดที่ 61 :** เช็คว่า playerInput เป็นการขึ้นบรรทัดใหม่หรือไม่ ถ้าใช่ให้ทำคำสั่งในบรรทัดที่ 63 – 65 แต่ถ้าไม่ใช่ให้ทำคำสั่งตามบรรทัดที่ 67

**บรรทัดที่ 63 – 65 :** เช็คว่า playerAnwser เป็นตัวเลข และอยู่ระหว่าง 1 – 100 ไหม ถ้าใช้กำหนด endCheck = 1 เพื่อจบการจรวจสอบ แต่ถ้าไม่ใช่กำนหนด invalidInput = 1

**บรรทัดที่ 67 :** กำหนด playerAnwser = (playerAnwser \* 10) + (playerInput – ‘0’) เพื่อให้ได้คำตอบที่เป็นตัวเลขจำนวนนับ และไม่ใช่ค่า ASCIS

**บรรทัดที่ 69 – 71 :** ถ้าค่าที่กรอกมาไม่ใช่ตัวเลขและไม่ใช่การขึ้นบรรทัดใหม่ ให้กำหนด invalidInput = 1

**บรรทัดที่ 73 – 77 :** ถ้าค่า invalidInput เท่ากับ 1 ให้แสดงข้อความว่ากรอกข้อมูลไม่ถูกต้อง ให้กรอกตัวเลขระหว่าง 1 – 100 และกำหนด endCheck = 1 เพื่อจบการตรวจสอบ



**บรรทัดที่ 79 :** ลบค่าที่ค้างอยู่ใน buffer

**บรรทัดที่ 81 :** จบ do while ทำงานซ้ำเมื่อ invalidInput เท่ากับ 1

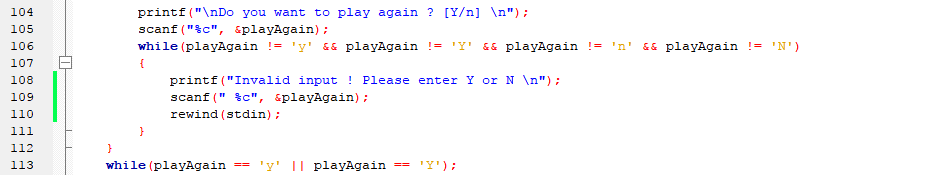
**บรรทัดที่ 83 - 85 :** ถ้าค่าที่ผู้เล่นกรอกเท่ากับคำตอบ ให้กำหนด win เท่ากับ 1

**บรรทัดที่ 86 – 88 :** ค่าที่ผู้เล่นกรอก น้อยกว่า คำตอบ แสดงข้อความว่า ตัวเลขที่กรอกมามีค่าน้อยไป

**บรรทัดที่ 89 – 92 :** ค่าที่ผู้เล่นกรอก มากกว่า คำตอบ แสดงข้อความว่า ตัวเลขที่กรอกมามีค่ามากไป

**บรรทัดที่ 94 :** จบ do while ทำงานซ้ำเมื่อ รอบทีทายไปแล้ว < 7 และ win เท่ากับ 0

**บรรทัดที่ 96 – 101 :** ถ้า win เท่ากับ 1 เมื่อเป็นจริง แสดงข้อความว่าผู้เล่นชนะในรอบที่เท่าไร เมื่อเป็นเท็จ แสดงข้อความว่าผู้เล่นแพ้และเฉลยคำตอบ



**บรรทัดที่ 103 :** ถามผู้เล่นว่าต้องการเล่นซ้ำหรือไม่

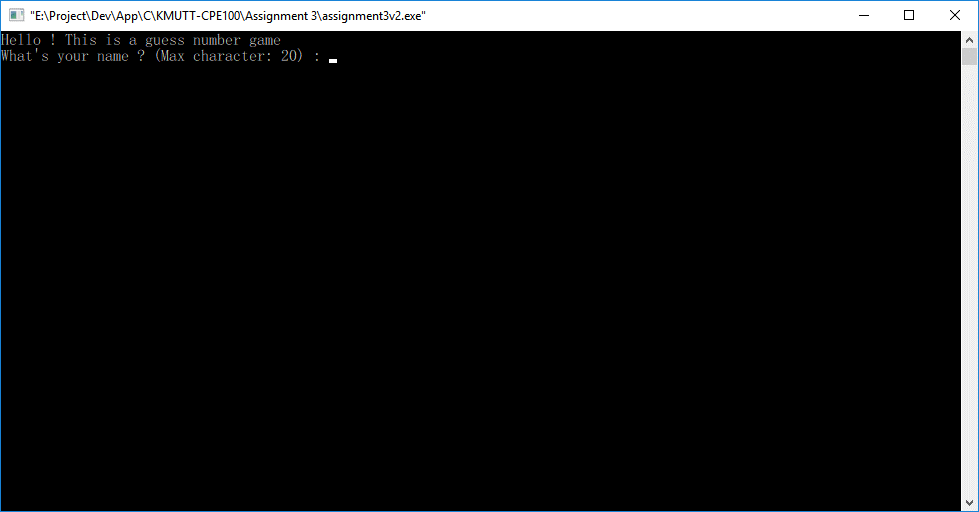
**บรรทัดที่ 104 :** รับค่าที่ผู้เล่นกรอกมา

**บรรทัดที่ 105 – 109 :** ตรวจสอบว่าค่าที่ผู้เล่นกรอกมา ถ้าไม่ใช่ y, Y, n และ N ให้แสดงข้อความว่ากรอกข้อมูลไม่ถูกต้องและให้ผู้เล่นกรอกค่ามาใหม่

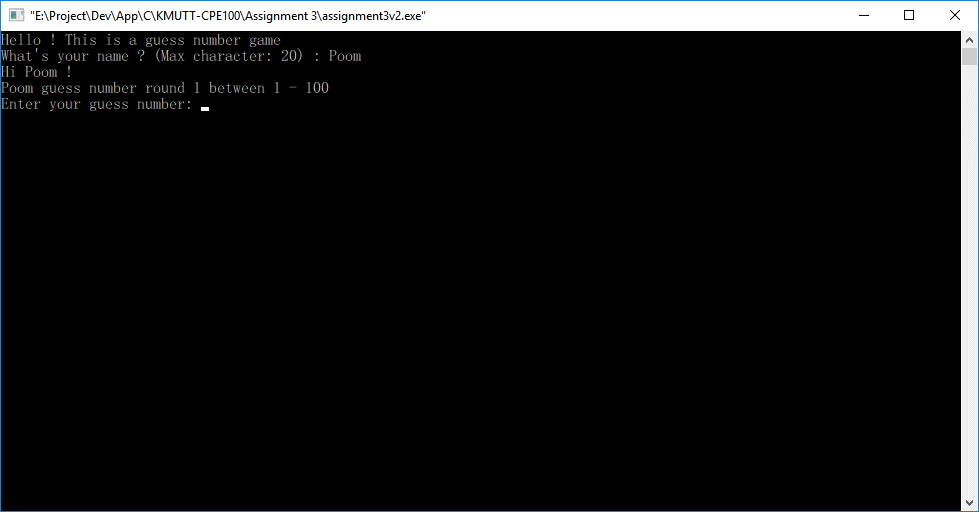
**บรรทัดที่ 111 :** จบ do while ทำงานซ้ำเมื่อค่าที่ผู้เล่นกรอกมาเท่ากับ y หรือ Y

**ผลการทำงาน**

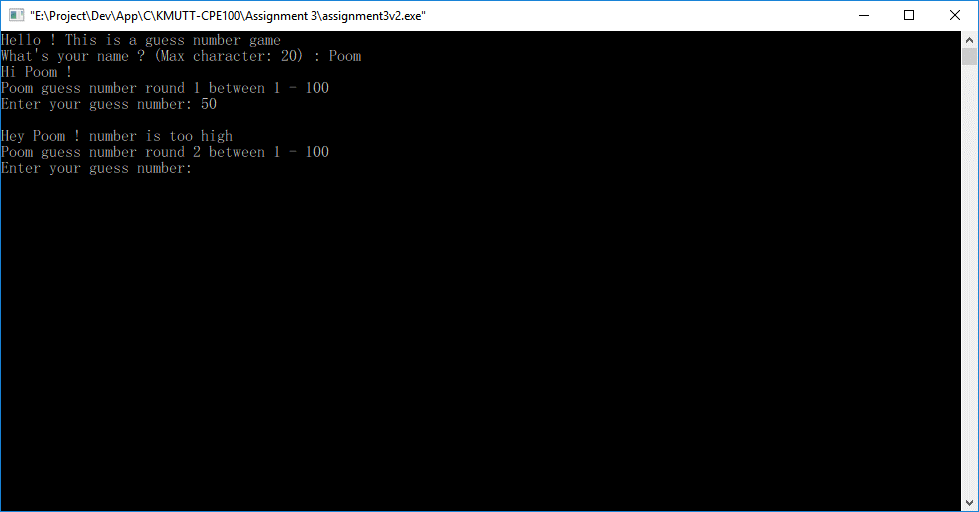
1. โปรแกรมกล่าวแนะนำตัวเอง



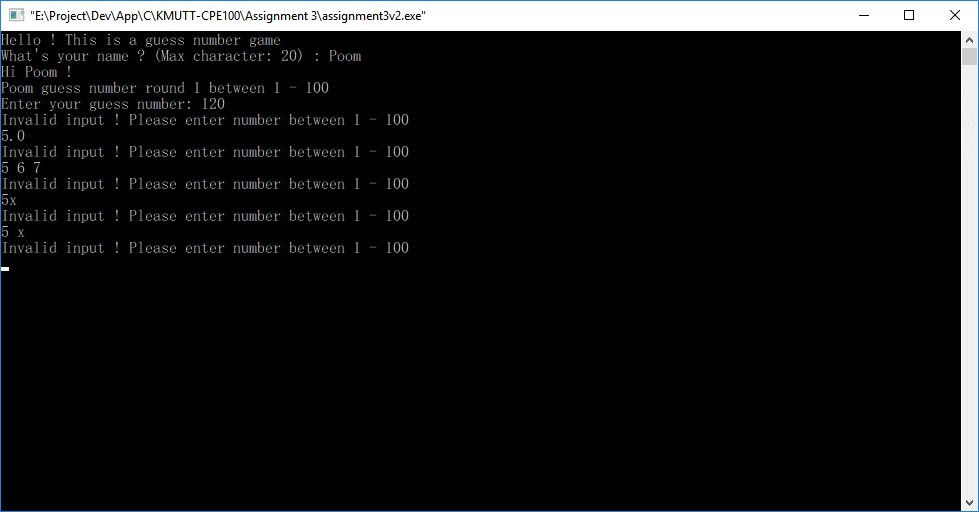
2. โปรแกรมถามชื่อผู้เล่นและมีการทักทาย



3. โปรแกรมกล่าวชื่อผู้เล่นทั้งในคำถามและคำตอบ



4. โปรแกรมสามารถป้องกันความผิดพลาดจากการป้อนข้อมูลค่าที่ไม่ต้องการได้



5. โปรแกรมมีคำถามว่าจะเริ่มเล่นใหม่หรือไม่  
 5.1 ) โปรแกรมรับเฉพาะปุ่ม y, Y, n, N เท่านั้น เมื่อกดผิดรอรับใหม่

5.2 ) หากกด y, Y โปรแกรมเริ่มเล่นเกมใหม่อีกครั้ง

5.3 ) หากกด n, N โปรแกรมจบการทำงาน

