

# การคำนวณความน่าจะเป็นจากมอนตีฮอลล์เกตส์ Probability Calculator from Monty Hall Gates.

เคมีชาติ วลัยภรณ์

Kemmachat Wanlayaphorn

Department of computer science, Faculty of Applied Science King Mongkut's University of Technology  
NorthBangkok

Email : s6004062636041@email.kmutnb.ac.th

## บทคัดย่อ

### ปริศนามอนตีฮอลล์ (Monty Hall Problem)

เป็นหนึ่งในปัญหาเรื่องความน่าจะเป็นที่ยากที่สุดในโลกในศตวรรษที่ 19 แม้ว่าปัญหานี้จะเข้าใจได้ไม่ยากนัก แต่ในศตวรรษที่ 19 ปัญหานี้ได้เป็นที่โด่งดังบนวารสาร The New York Times เนื่องจากปัญหานี้ทำให้หลายคนแม้แต่ นักคณิตศาสตร์ระดับโลกนั้นตอบผิดมาแล้ว สำหรับโปรเจกต์นี้เป็นการเสนออัลกอริทึม คำนวณค่าความน่าจะเป็นของมอนตีฮอลล์เกตส์ด้วยภาษาซี รวมถึงการประยุกต์ความน่าจะเป็นในสถานการณ์ต่างๆที่จำนวนเฉลย  $w_0$  ต่างกัน บนกฎจากโฮสที่อนุญาตให้มีประตูที่ถูกเพียงบานเดียว

**คำสำคัญ :** มอนตีฮอลล์ ความน่าจะเป็น

## บทที่ 1

### เนื้อหา

ปัญหาข้อนี้ถูกตั้งตามชื่อ มอนตี ฮอลล์ ผู้ดำเนินรายการเกมโชว์อเมริกันชื่อ Let's Make a Deal ซึ่งมีรูปแบบดังนี้ ในรายการจะมีประตูสามบาน หนึ่งในด้านหลังของประตูนั้นมีรถยนต์ อีกสองบานมีแพะ ผู้ร่วมรายการต้องเลือกประตูหนึ่งบาน ซึ่งถ้าโชคดีก็จะได้รถยนต์กลับบ้านไป สมมติว่าเลือกประตูเบอร์ 1 จากนั้นเพื่อให้ลุ้นมากขึ้น ผู้ดำเนินรายการจะเดินมาเปิดประตูสักบานที่มีแพะอยู่ สมมติว่าเป็นประตูเบอร์ 3 แน่นอนว่ามอนตี ฮอลล์ ผู้เป็นพิธีกรนั้นรู้ดีว่าประตูบานไหนมีรถยนต์ และประตูบานไหนมีแพะ ดังนั้นไม่ว่าเราเลือกประตูบานไหน เขาก็จะเดินมาเปิดประตูบานที่มีแพะได้เสมอ จากนั้นพิธีกรจะถามผู้ร่วมรายการว่าจะเปลี่ยนประตูที่เลือกไว้หรือไม่? ผู้ร่วมรายการบางคนก็เปลี่ยนประตูที่เลือกไว้ บางคนก็ยืนยันเลือกประตูบานเดิมเชื่อใหม่ว่า หากเราเปลี่ยนประตูที่เลือกไว้ หลังจากพิธีกรเดินมาเปิดประตูแพะ จะทำให้โอกาสได้รถยนต์กลับบ้านสูงกว่าการไม่เปลี่ยนประตู!

คำเฉลยของปัญหามอนตี ฮอลล์ มีหลายรูปแบบหลายเวอร์ชัน แต่ผมจะเล่าให้ฟังสองแนวคิด วิธีคิดแรกคือ การแจกแจงรูปแบบที่เป็นไปได้โดยตรงไปตรงมา

กล่าวคือ เมื่อเราเลือกประตูบานที่ 1

รูปแบบที่เป็นไปได้มี 3 กรณี

กรณีที่ 1 คือ หลังประตูบาน 1 เป็นรถยนต์

กรณีที่ 2 คือ หลังประตูบาน 2 เป็นรถยนต์

กรณีที่ 3 คือ หลังประตูบาน 3 เป็นรถยนต์

ดังนั้นเมื่อพิธีกรเปิดประตูพะไปแล้วหนึ่งบาน การเลือกประตูบานเดิมแล้วมีโอกาสได้รถยนต์มีเพียงกรณีเดียวคือ กรณี 1 แต่ถ้าเปลี่ยนประตู โอกาสจะได้รถยนต์มีสองกรณีคือ กรณี 2 และ 3

ดังนั้นการไม่เปลี่ยนประตูจึงมีโอกาสได้รถยนต์เพียง  $1/3$  ส่วนการเปลี่ยนประตูนั้นมีโอกาสได้รถยนต์  $2/3$

วิธีคิดที่สองคือ การมองว่า การเลือกประตูบานแรกมีโอกาสได้รถยนต์เพียง  $1/3$  โอกาสในการได้รถยนต์อีก  $2/3$  อยู่ที่ประตูทั้งสองที่เหลือ

การเปิดประตูบานที่ 3 ออกไปหนึ่งบาน ทำให้โอกาสในการเจอรถยนต์หลังประตูบานนั้นเป็น 0 แต่ประตูอีกบานที่ยังไม่ได้เลือกก็ยังมีโอกาสเป็น  $2/3$  เช่นเดิม

### ประโยชน์ที่ได้รับ

1. ทำให้เราเข้าใจว่ากลไกของความน่าจะเป็นแท้จริงแล้วขึ้นอยู่กับเงื่อนไขที่เกิดขึ้น
2. เป็นประโยชน์ต่อผู้ที่สนใจ/ศึกษาเพื่อนำไปต่อยอดในอนาคต

### ทางแยกจากการเปลี่ยนกฎของไฮส

สำหรับปัญหามอนตีฮอลล์เกตส์ ผลเฉลี่ยของความน่าจะเป็นที่หลังประตูจะมีรถ  
เมื่อกฎของไฮสอนุญาตให้มีการเปลี่ยนประตูมีค่าเท่ากับ  $\frac{2}{3}$  จากทฤษฎี  
แต่หากเป็นในสถานการณ์ที่ต่างกันที่ผลเฉลี่ยจำนวนประตู ( $w_0$ ) หรือจำนวนประตูที่เพิ่มขึ้น ( $n$ )  
ผลเฉลี่ยของความน่าจะเป็นจะเปลี่ยนแปลงอย่างไร

จากการวิเคราะห์หากเราเปลี่ยนกฎของไฮส 2 กรณีโดยขึ้นกับจำนวนประตูที่เฉลี่ย ( $w_0$ ) และจำนวนประตู ( $n$ )  
ในกรณีแรกกฎของไฮสจะอนุญาตให้มีการประตูที่ถูกเพียงบานเดียวและเฉลี่ยประตูที่ผิดอย่างต่อเนื่องที่ ( $w_0 < n - 2$ )  
ในกรณีนี้จะมีความน่าจะเป็นที่เรียกว่า “ ความน่าจะเป็นต่อเนื่องไม่อิสระ ”  
ความต่อเนื่องนี้มาจากการเฉลี่ยประตูที่ผิดอย่างต่อเนื่องภายใต้กฎของไฮสและไม่อิสระจากการที่ไฮสให้ผู้เล่นเกมสามารถเลือก  
ประตูและเปลี่ยนภายใต้กฎของไฮสได้เพียงบานเดียว และ ในกรณีที่สอง คือ  
ความน่าจะเป็นบนกฎของไฮสที่อนุญาตให้มีการประตูที่ถูกเพียงบานเดียวและสามารถสลับประตูได้หลังจากเฉลี่ยทุกครั้ง  
( $w_0 < n - 2$ ) เรียกว่า “ ความน่าจะเป็นไม่ต่อเนื่องอิสระ ”

### ความน่าจะเป็นต่อเนื่องไม่อิสระ

ความน่าจะเป็นต่อเนื่องไม่อิสระ คือ  
ความน่าจะเป็นบนกฎของไฮสที่อนุญาตให้มีการประตูที่ถูกเพียงบานเดียวและเฉลี่ยประตูที่ผิดอย่างต่อเนื่องที่ ( $w_0 < n - 2$ )  
สมการต่อเนื่องไม่อิสระ

$$P_n = \left[ \left( \frac{n_s}{n_s + 1} \right) (\beta) \right]$$

$$\text{ที่ } \beta = \frac{1}{n_s - w_0}$$

เมื่อ  $\beta$  คือความน่าจะเป็นบนกรอบเขี่ยนี้

\*\*\* เนื่องจาก  $\beta$  เป็นตัวแปรที่ถูกสมมุติขึ้นมาจากการให้นิยามว่า  
 “สำหรับความน่าจะเป็นใดๆที่ไม่ขึ้นอยู่กับการเปลี่ยนแปลงจะมีค่าคงที่ของความน่าจะเป็น เช่น การหยิบสุ่ม”  
 \*\*\*

จาก  $\beta = \frac{1}{n_s - w_0}$  ที่เป็นฟังก์เตอร์ตัวคูณเมื่อแทนค่าตัวแปรใดๆเข้าสู่สมการนั้นได้พบว่าเป็นอนุกรมลู่เข้าสู่ 0  
 ทำให้เราสามารถเขียนเป็นอนุกรมได้ดังนี้ เมื่อ  $(w_0 < n - 2)$

$$\beta = \frac{1}{2} \times \frac{2}{3} \times \frac{3}{4} \times \frac{4}{5} \times \frac{5}{6} \times \frac{6}{7} \times \dots \dots \times \frac{n}{n+1} \text{ ที่ } n_s \geq 3$$

จะได้ว่า

$$\beta = \prod_{i=1}^{n-3} \left( \frac{i}{i+1} \right)$$

$$P_n = \left[ \left( \frac{n_s}{n_s + 1} \right) \prod_{i=1}^{n-3} \left( \frac{i}{i+1} \right) \right]$$

$P_n$  คือความน่าจะเป็นที่แปรผันกับเงื่อนไข

$n_s$  คือจำนวนประตูที่ไม่ถูกเลือก เมื่อ  $n_s + n'_s = n$

$$\text{และ } n'_s = 1$$

โดยที่  $n'_s$  คือประตูที่ถูกเลือกจากกฎของไฮสที่ให้เลือกประตูได้เพียงบานเดียว

โอกาสที่แปรผันตามกับเงื่อนไขเมื่อพิจารณาร่วมกับโอกาสที่สูญเสีย

$$P(n_s) \sim (100 - P_l) \left[ \left( \frac{n_s}{n_s + 1} \right) \left( \frac{1}{n_s - w_0} \right) \right]$$

หรือ

$$P(n_s) \sim (100 - P_l) \left[ \left( \frac{n_s}{n_s + 1} \right) \prod_{i=1}^{n-3} \left( \frac{i}{i+1} \right) \right]$$

ที่  $n_s \geq 3$

$P_i$  คือโอกาสที่สูญเสีย

ความน่าจะเป็นไม่ต่อเนื่องอิสระ

ความน่าจะเป็นไม่ต่อเนื่องอิสระ คือ

ความน่าจะเป็นบนกฎของไฮสที่อนุญาตให้มีประตูที่ถูกเพียงบานเดียวและสามารถสลับประตูได้หลังจากเฉลยทุกครั้งที่  
 $(w_0 < n - 2)$

สมการไม่ต่อเนื่องอิสระ

สมการลำดับที่หนึ่ง

$$\sum P_n = \sum_{1 \leq w_0 < n_s} \left[ \left( \frac{n_s}{n_s + 1} \right) \left( \frac{1}{n_s - w_0} \right) \right] + \frac{1}{n} = 1$$

สมการลำดับที่สอง

$$\left( \sum P_{nA} + |Pw_0| \right) = 1$$

$$P_{nAi} = P_{nA} + \frac{Pw_0}{n_s}$$

$P_{nA}$  คือความน่าจะเป็นที่แปรผันกับเงื่อนไขของประตูแต่ละบานหลังจากเฉลย  $w_0$  ที่สมการลำดับที่หนึ่ง

$Pw_0$  คือความน่าจะเป็นที่สูญเสียจากการเฉลย  $w_0$  ของไฮส

$P_{nAi}$  คือความน่าจะเป็นที่แปรผันกับเงื่อนไขของประตูแต่ละบานหลังจากเฉลย  $w_0$  จากสมการลำดับที่สอง

สมการลำดับที่สาม

$$LP_n = (Lw_0) + (Ln'_s)$$

$LP_n$  คือความน่าจะเป็นสุดท้ายที่แปรผันกับเงื่อนไข

$Lw_0$  คือความน่าจะเป็นสุดท้ายจากการเฉลี่ย  $w_0$  ของโฮสต์

$Ln'_s$  คือความน่าจะเป็นสุดท้ายจากประตูที่ถูกเลือก

## บทที่ 2

### วิธีการใช้งานโปรแกรม

"C:\Users\aut\_1\Documents\project done\project\Real\_project.exe"

```
This program create for solve monty hall gate

The Monty Hall problem is a brain teaser, in the form of a probability puzzle,
loosely based on the American television game show

Suppose you're on a game show, and you're given the choice of three doors: Behind one door is a car; behind the other is goats.
At first you pick some door of all
Second the host who knows what's behind the doors will open some door of goats
Then the host give the question to you (You have a chance Do you wanna change your choice ?)

Start the program (y/n): _
```

1. เมื่อเริ่มเปิดโปรแกรมขึ้นมาจะพบกับ Intro ที่เป็นการกล่าวนำถึงความเป็นมาของ Monty Hall Gate และจะพบคำถามเพื่อเป็นการตอบเริ่มต้นโปรแกรม โดย พิมพ์ y เพื่อดำเนินการเข้าสู่โปรแกรม

"C:\Users\aut\_1\Documents\project done\project\Real\_project.exe"

This program create for solve monty hall gate

The Monty Hall problem is a brain teaser, in the form of a probability puzzle, loosely based on the American television game show

Suppose you're on a game show, and you're given the choice of three doors: Behind one door is a car; behind the other is goats.

At first you pick some door of all

Second the host who knows what's behind the doors will open some door of goats

Then the host give the question to you (You have a chance Do you wanna change your choice ?)

Start the program (y/n): y

How many your gate ? (recommend 3-4) : \_

2.หลังจากที่ได้ทำการเปิดโปรแกรมเป็นที่เรียบร้อยแล้วก็จะพบกับคำถามที่ว่าคุณต้องการที่ประตู ให้ใส่จำนวนระหว่าง (3-4) เนื่องจาก  $n$  ที่เป็นจำนวนประตู เข้าสู่นั่นยังคงเป็นแนวทางการพัฒนาโปรแกรมในอนาคตข้างหน้า ผู้พัฒนาตั้งใจทำรูปแบบนี้เพื่อเป็นการปูทางในการพัฒนาโปรแกรมต่อไป (ถ้าใส่จำนวน  $n$  ที่มากกว่า 4 อาจจะส่งผลให้การแสดงผล รูปแบบ 200 ตัวแรกคลาดเคลื่อน แต่ยังคงสามารถหา ความน่าจะเป็นได้อย่างแม่นยำ)

"C:\Users\aut\_1\Documents\project done\project\Real\_project.exe"

This program create for solve monty hall gate

The Monty Hall problem is a brain teaser, in the form of a probability puzzle, loosely based on the American television game show

Suppose you're on a game show, and you're given the choice of three doors: Behind one door is a car; behind the other is goats.

At first you pick some door of all

Second the host who knows what's behind the doors will open some door of goats

Then the host give the question to you (You have a chance Do you wanna change your choice ?)

Start the program (y/n): y

How many your gate ? (recommend 3-4) : 3

How many door was solve ? : \_

3.เมื่อใส่จำนวนประตูเป็นที่เรียบร้อยแล้วจะมีคำถามต่อมาว่าผู้ใช้ต้องการที่เฉลยที่ประตู โดยมีข้อแม้ว่า

$(w_0 < n - 2)$  ตามเบื้องต้นที่ได้กล่าวไว้ใน บทที่ 1

"C:\Users\aut\_1\Documents\project done\project\Real\_project.exe"

How many door was slove ? : 1

Switch by Theory

Result from the 200 first of all

1 = game win 0 = game lose

0	0	1	0	1	0	1	1	1	0
0	1	0	1	0	1	1	1	1	1
1	1	1	1	1	0	1	1	1	0
1	1	1	1	0	1	1	1	0	1
1	1	1	1	1	0	1	1	1	1
1	1	0	1	1	0	1	0	1	0
1	1	0	1	1	1	0	0	1	1
1	0	0	1	1	1	1	1	1	1
1	1	1	1	0	1	1	0	0	1
1	1	1	1	1	1	1	1	1	0
0	1	0	1	1	1	1	1	1	0
1	1	1	0	1	0	0	0	0	0
0	1	0	1	1	1	1	1	1	0
1	0	1	1	1	1	0	0	0	1
1	0	1	1	1	1	0	1	0	0
1	0	1	1	1	1	1	1	1	0
1	0	1	1	1	1	1	1	1	0
1	0	1	1	1	1	1	1	1	0
1	1	1	0	1	0	1	0	1	0
1	1	0	0	1	1	1	1	1	0
1	1	1	1	1	1	1	1	1	0
1	1	1	1	1	1	1	1	1	0
1	1	1	1	0	1	1	0	1	0
1	1	0	0	1	1	1	1	1	0
1	1	1	1	1	1	1	1	1	1

66.679600 percent win rate  
we win 333398 and we lose 166602

"C:\Users\aut\_1\Documents\project done\project\Real\_project.exe"

Without Theory

Result from the 200 first of all

1 = game win 0 = game lose

0	0	0	0	0	0	0	0	1	0
0	0	0	1	0	0	1	0	0	0
0	0	0	1	0	0	0	0	0	1
1	0	1	1	1	0	0	1	0	0
1	1	0	0	1	1	0	1	0	0
0	1	0	0	0	0	0	0	1	1
0	0	1	1	0	0	0	0	0	1
1	0	0	0	0	0	0	1	1	0
0	0	1	0	0	0	0	1	0	0
1	1	1	0	0	1	0	0	1	0
0	0	1	0	0	0	0	0	0	0
0	1	0	0	0	0	1	0	1	0
0	0	1	0	1	0	0	0	0	0
1	1	0	1	1	0	1	0	1	0
1	1	0	0	0	0	1	0	1	1
0	0	0	0	0	0	0	0	0	1
0	1	0	0	0	0	1	0	1	0
1	0	1	1	0	0	0	1	1	0
1	1	1	1	0	1	0	0	0	1
0	0	0	0	1	1	1	0	0	0

33.380600 percent win rate  
we win 166903 and we lose 333097

Win by Theory 333398 game

Win by without Theory 166903 game

166495 game win is result of comparison test between probability with situation probability.  
we know that a chance at change will give us more win

Process returned 0 (0x0) execution time : 1289.089 s  
Press any key to continue.

4. เมื่อได้ข้อมูลจำครบ โปรแกรมจะนำข้อมูลทั้งหมดไป Process 500,000 ครั้ง และทำฟังก์ชันต่างๆ เพื่อนำมาแสดงผล



ทั้งแบบใช้ทฤษฎี และไม่ใช้ทฤษฎี จนรวมถึงกับเปรียบเทียบกัน และหาความแตกต่างระหว่างแบบใช้ทฤษฎีกับไม่ใช้ทฤษฎี

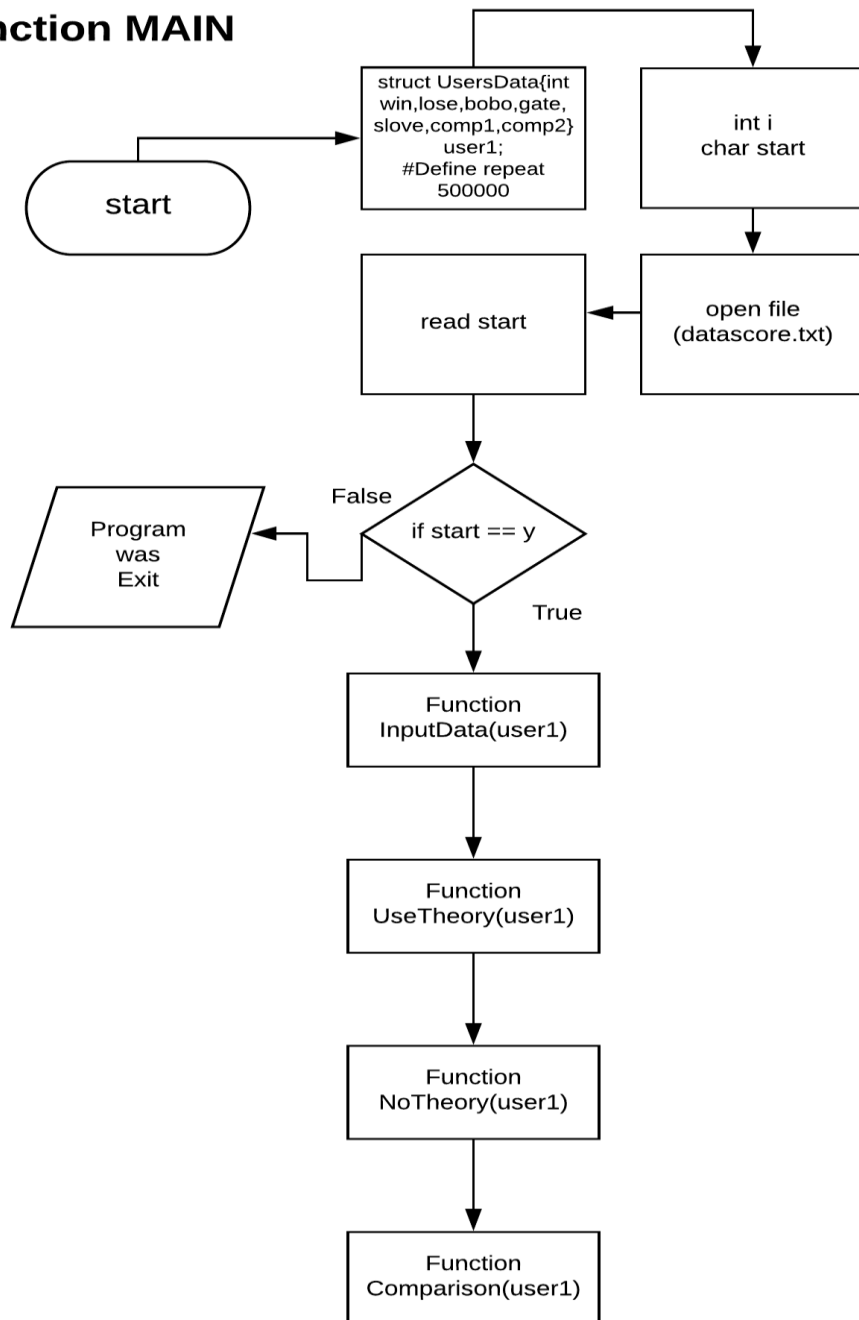
### บทที่ 3

ฟังก์ชันที่ใช้ :

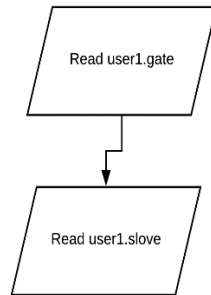
Void InputData(ak);	ใช้เก็บข้อมูลที่ใช้ต้องการ
Void UseTheory(ut);	ใช้เพื่อพิสูจน์แบบทฤษฎี
Void NoTheory(nt);	ใช้เพื่อพิสูจน์แบบไม่ใช้ทฤษฎี
Void Comparison(cp);	ใช้เพื่อเปรียบเทียบค่าความต่างของความน่าจะเป็น

### Flowchart

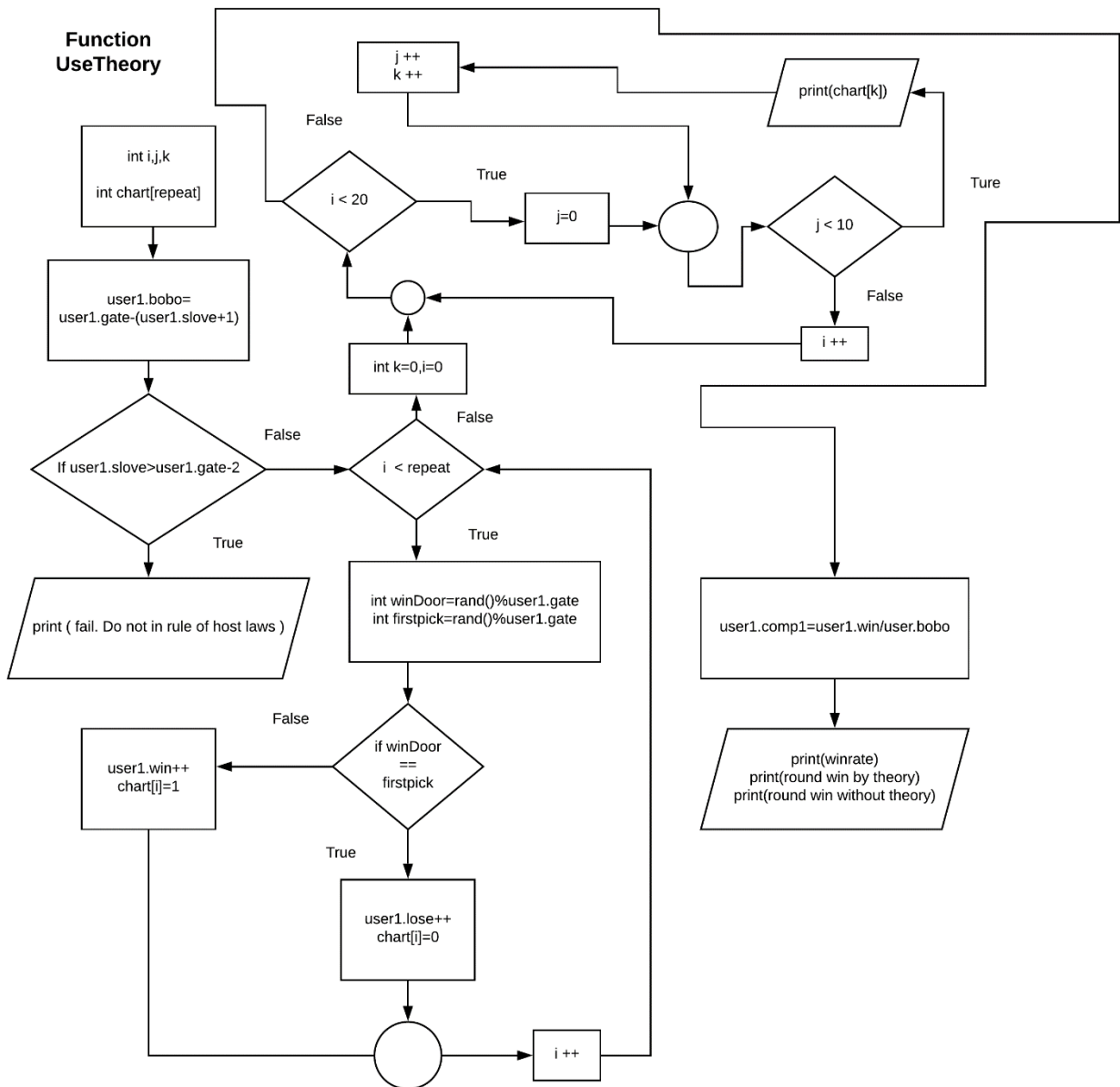
## Function MAIN



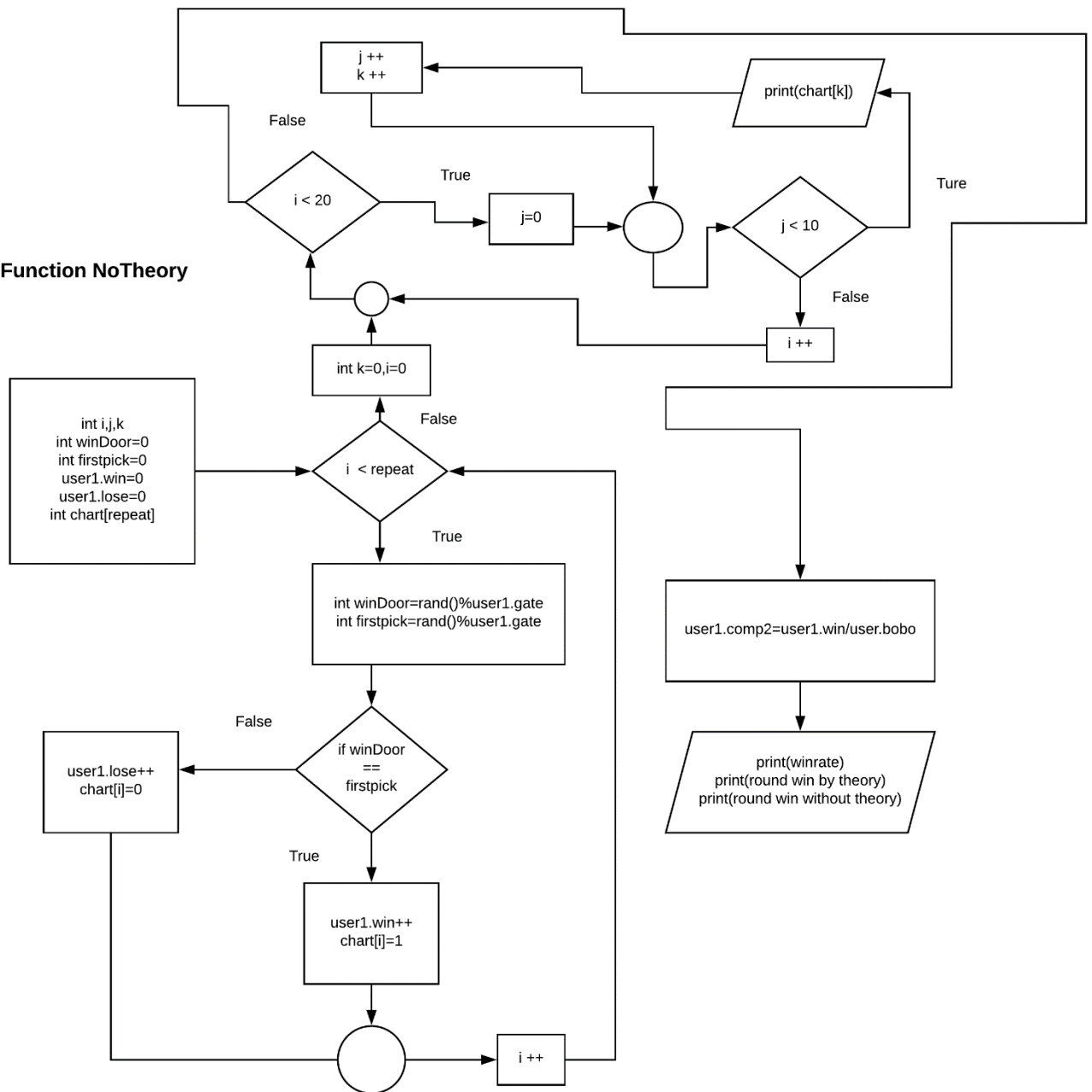
### Function InputData



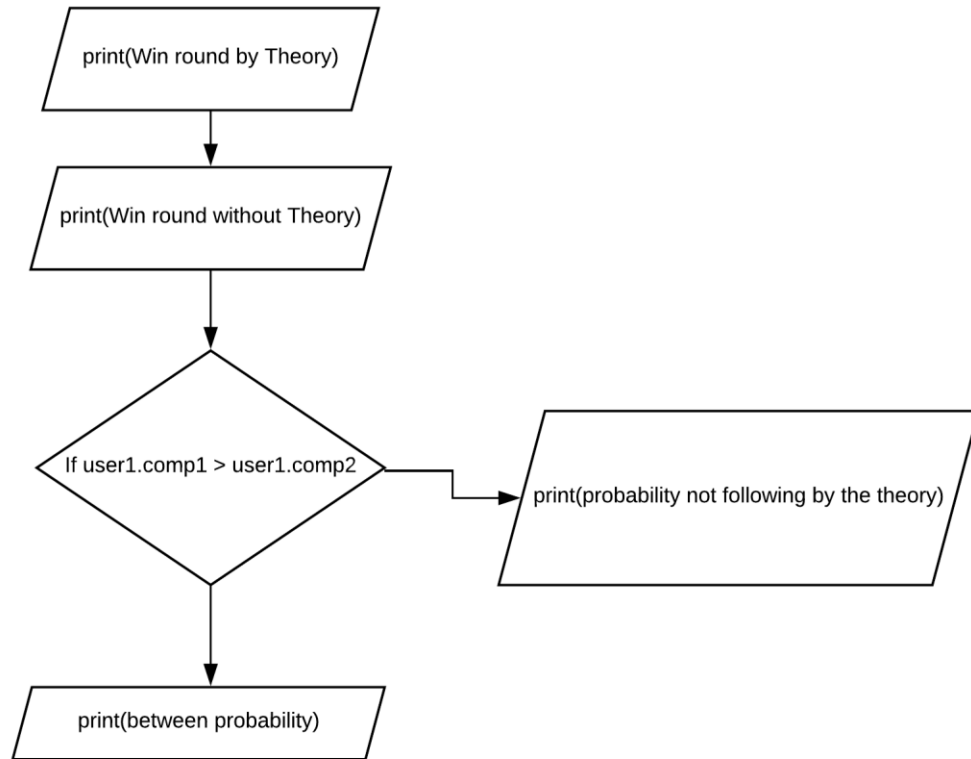
### Function UseTheory



Function NoTheory



### Function Comparison



### บทที่ 4

#### แนวทางการพัฒนา

จากการทำโปรเจกต์เพื่อพิสูจน์ความน่าจะเป็นที่เปลี่ยนแปลงจากเงื่อนไขแล้วทำให้ผู้ทำโครงการได้สมการที่ใช้คำนวณหาความน่าจะเป็นเมื่อเปลี่ยนแปลงเงื่อนไขไปในกรณีที่ **1** ที่เข้าสู่ค่าอนันต์ได้สำเร็จ สำหรับแนวทางการพัฒนาผู้ทำโครงการได้วางแนวทางการพัฒนาที่ความน่าจะเป็นกรณีที่ **2** ซึ่งมีความซับซ้อนมากขึ้น