1. **最後需交出一個 function 叫 craps\_game ， 該function有兩個輸入：num\_simulations (the number of simulations to run) 和 bet\_size (the size of the bet for each game)。function裡為模擬此賭博遊戲的輸贏。**

# 依照題意列出函數結構

# craps\_game <- function(num\_simulations, bet\_size) { }

1. **此賭博遊戲為，用sample()去模擬擲兩個骰子。加總擲出的兩個骰子數字，為點數和。**

# 建立一個function叫point，會回傳加總值出的兩個骰子點數和

# 透過sample(1:6,2)隨機取得兩個1~6之間的整數，並透過sum()將sample取得的數字做加總

point <- function() {

return (sum(sample(1:6,2)))

}

# 執行多次point()的結果，應為2~12常態分布



1. **如果第一次點數和為 7 或 11，玩家勝，贏得 bet size 。 若點數和為 2、3 或 12，玩家輸，輸了 -1 x bet size。如果點數和為其他點數，記錄第一次的點數和，然後繼續擲骰，直至點數和等於第一次擲出的點數和，玩家勝，贏得 bet size。若在這之前擲出了點數和為 7，玩家輸，輸了 -1 x bet size。**

# 建立一個function叫if\_win，依照上述規則判斷是否贏得bet\_size

if\_win <- function() {

# 先透過前面寫的piont()取得第一次點數和

first\_point <- point()

# 如果點數和為7或11，回傳TRUE代表贏得遊戲

if (first\_point == 7 || first\_point == 11) {

return (TRUE)

# 如果點數和為2、3或12，回傳FALSE代表輸了

} else if (first\_point == 2 || first\_point == 3 || first\_point == 12) {

return (FALSE)

} else {

# 如果為其他點數，先記錄第一次點數和

target <- first\_point

retry\_point <- 0

# 繼續擲骰直到點數和等於第一次或7

while (retry\_point != 7 && retry\_point != target) {

retry\_point <- point()

}

# 如果等於第一次和，回傳TRUE代表贏得遊戲

if (retry\_point == target) {

return (TRUE)

# 如果等於7和，回傳FALSE代表輸了

} else if (retry\_point == 7) {

return (FALSE)

}

}

}



1. **計算遊戲的期望值  
   expected value = (probability of winning x bet size) + (probability of losing x bet size).**

# 建立一個function叫expected\_value，需傳入贏的機率和賭注金額兩個參數

# 透過probability\_of\_winning\*bet\_size取得贏的期望值

# 只有輸贏兩種可能，所以透過1 - probability\_of\_winning取得輸的機率

# 輸的金額是負的bet\_size，所以贏的期望值減輸的期望值就是整個遊戲的期望值

# 可以再簡化成return ((2\*probability\_of\_winning-1)\*bet\_size)

# 但考量到便於理解，以及可能需要調整輸贏結果，不再做進一步簡化

expected\_value <- function(probability\_of\_winning, bet\_size) {

return (probability\_of\_winning\*bet\_size - (1-probability\_of\_winning)\*bet\_size)

}

# 放機率0.6跟賭注金額10，用來測試函數



1. **craps\_game function 輸出（Return） 遊戲的期望值。**

**提示：先寫玩一次的function，贏為TRUE，輸為FALSE，輸出該結果。再放進craps\_game重複多次後，計算輸贏次數，獲得輸贏機率。**

# 建立一個function叫craps\_game，如第一題所述需傳入測試次數和賭注金額兩個參數

craps\_game <- function(num\_simulations, bet\_size) {

# 建立一個變數叫win\_count

win\_count­ <- 0

# 依照傳入參數重複執行if\_win()，如果回傳值為正，win\_count+1

for (i in c(1:num\_simulations)) {

if (if\_win()) {

win\_count <- win\_count + 1

}

}

# 贏的機會 = 前面算出來的win\_count除以總執行次數

probability\_of\_winning <- win\_count/num\_simulations

# 透過前面寫的expected\_value來取得期望值

return(expected\_value(probability\_of\_winning, bet\_size))

}

# 放執行1000次跟賭注金額10，用來測試函數

