3.4 節末問題 3.4 的解答

問題 3.4.1

21 個模式不一定會等機率發生。事實上是：

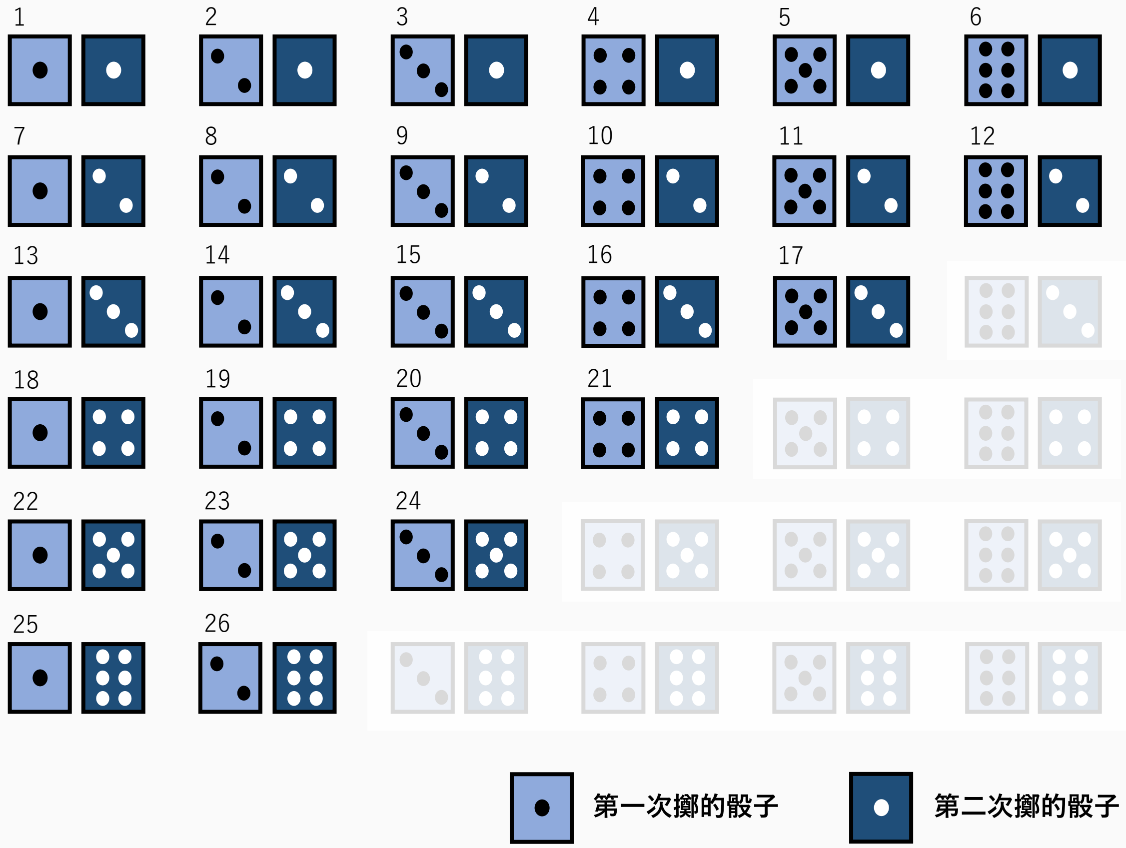
- 出現 $(1, 1)$ 的機率是 $1/36$
- 出現 $(1, 2)$ 的機率是 $1/18$

由於這些機率不同，不能單純地認為「21 個模式中有 15 個模式的點數和為8以下，所以所求的機率是 15/21」。

另外，用 ★ 表示的部分可以理解如下。

。

> 擲兩個骰子時，出現 $(1, 2)$ 點數的模式有
>
> - 第一次擲出的是 1，第二次擲出的是 2
> - 第二次擲出的是 2，第一次擲出的是 1
>
> 這 2 種情況。由於在 36 種情況中佔 2 種，機率是 $2/36 = 1/18$。



問題 3.4.2

將期望值公式（→**3.4.2項**）代入，答案如下：

$$\left(1000000\times\frac{1}{10000}\right)+\left(100000\times\frac{9}{10000}\right)+\left(10000\times\frac{9}{1000}\right)+\left(1000\times\frac{9}{100}\right)+\left(0\times\frac{9}{10}\right)$$

$$= 100 + 90 + 90 + 90$$

$$= 370$$

由於獲得獎金的期望值是 370 日圓，當參加費是 500 日圓的情況下是虧損的。

問題 3.4.3

首先，由於期望值的線性關係（→**3.4.3項**），以下關係成立。

（學習時間的期望值總和）

＝（第 1 天學習時間的期望值）＋ ⋯ ＋（第 N 天學習時間的期望值）

因此，每天的學習時間的期望值如下：

- 第 1 天：$(A_1\times1/3)+(B_1\times2/3)$
- 第 2 天：$(A_2\times1/3)+(B_2\times2/3)$
- 第 3 天：：$(A_2\times1/3)+(B_2\times2/3)$
- ：
- 第 N 天：$(A_2\times1/3)+(B_2\times2/3)$

因此，所求的總學習時間的期望值如下：

$$\left(A_1\times\frac{1}{3}+B_1\times\frac{2}{3}\right)+\left(A_2\times\frac{1}{3}+B_2\times\frac{2}{3}\right)+\cdots+\left(A_N\times\frac{1}{3}+B_N\times\frac{2}{3}\right)$$

撰寫將此值輸出的程式即為正解。以下是 C++的解答範例。

```
#include <iostream>
using namespace std;

int N;
double A[109], B[109];
double Answer = 0.0;

int main() {
```

下一頁

```
    // 輸入
    cin >> N;
    for (int i = 1; i <= N; i++) cin >> A[i] >> B[i];

    // 求出期望值
    for (int i = 1; i <= N; i++) {
        double eval = A[i] * (1.0 / 3.0) + B[i] * (2.0 / 3.0);
        Answer += eval;
    }

    // 輸出
    printf("%.12lf\n", Answer);
    return 0;
}
```

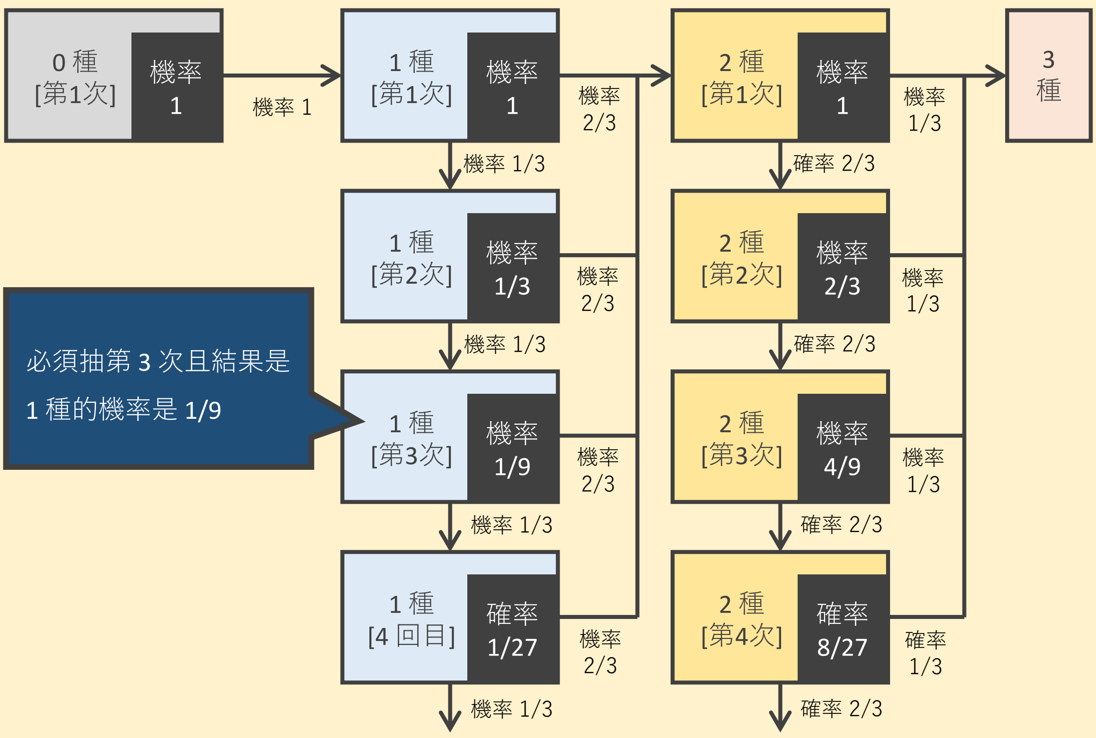
※ Python等原始碼請參閱 chap3-4.md。

問題 3.4.4

此問題的設定很複雜且難度較高，因此先思考 $N = 3$ 的情況。

首先，表示收集到所有硬幣的過程的圖如下。

- 實線表示獲得了尚未收集過的硬幣種類
- 虛線表示抽到了已經收集過的硬幣種類



因此，由於期望值的線性關係，收集所有種類所需次數的期望值是以下總和。

- 從 0 種變成 1 種所需次數的期望值 ⋯ (1)
- 從 1 種變成 2 種所需次數的期望值 ⋯ (2)
- 從 2 種變成 3 種所需次數的期望值 ⋯ (3)

根據前一頁的圖及和的公式（→**2.5.10 項の最後**）：

- (1) 的期望值是 1 次
- (2) 的期望值是は $1 + (1/3) + (1/9) + \cdots = 3/2$次
- (3) 的期望值是 $1 + (2/3) + (4/9) + \cdots = 3$ 次

因此，當 $N = 3$時，答案是 $1 + 3/2 + 3 =$ **11/2 回** 次。。

接下來，思考一般 N 的情況。由於期望值的線性關係，所求的答案是以下值的總和。

- 從 0 種變成 1 種所需次數的期望值
- 從 1 種變成 2 種所需次數的期望值
- ：
- 從 $N - 1$種變成 N 種所需次數的期望值

因此，從已經收集到 r 種硬幣的狀態到收集到第 $r + 1$ 種硬幣時，拿到已經收集過的硬幣的機率是 r/N，所以：、

- 需要 1 次以上的機率： 1
- 需要 2 次以上的機率： $(r/N)^1$
- 需要 3 次以上的機率： $(r/N)^2$
- 需要 4 次以上的機率： $(r/N)^3$ [以下略]

因此，次數的期望值如下：

$$1 + \left(\frac{r}{N}\right)^1 + \left(\frac{r}{N}\right)^2 + \left(\frac{r}{N}\right)^3 + \cdots = \frac{N}{N - r}$$

最後，總次數的期望值是對 $r = 0,1,2,\ldots,N - 1$ 將紅色部分相加如下。

$$\frac{N}{N} + \frac{N}{N-1} + \frac{N}{N-2} + \cdots + \frac{N}{2} + \frac{N}{1}$$

因此，撰寫如下將此值輸出的程式即為正解。

```
#include <iostream>
using namespace std;

int N;
double Answer = 0;

int main() {
    // 輸入
    cin >> N;

    // 求出期望值
    for (int i = N; i >= 1; i--) {
        Answer += 1.0 * N / i;
    }

    // 輸出
    printf("%.12lf\n", Answer);
    return 0;
}
```

※ Python等原始碼請參閱 chap3-4.md。