

Q4_Sample_P2a_Recursive

Jacques Hadamard นักคณิตศาสตร์ชาวฝรั่งเศส ได้นิยาม Hadamard matrix เป็นเมทริกซ์สี่เหลี่ยมจัตุรัส ที่ประกอบด้วยเลข 1 และ -1 เท่านั้น Hadamard matrix สามารถสร้างได้จากความสัมพันธ์ $H_1 = [1]$ และ

$$H_n = \begin{bmatrix} H_{n-1} & H_{n-1} \\ H_{n-1} & -H_{n-1} \end{bmatrix} \text{ เพื่อให้เห็นภาพมากขึ้น ให้พิจารณาตัวอย่างต่อไปนี้}$$

$$H_2 = \begin{bmatrix} H_1 & H_1 \\ H_1 & -H_1 \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

$$H_3 = \begin{bmatrix} H_2 & H_2 \\ H_2 & -H_2 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix}$$

นอกจากนี้ Hadamard matrix ยังมีสมบัติต่าง ๆ ที่น่าสนใจ ให้นิสิตเขียนฟังก์ชันต่าง ๆ ที่เกี่ยวข้องกับ Hadamard matrix ดังนี้

H(n) เป็นฟังก์ชันที่คืน Hadamard matrix ลำดับที่ n (คืน matrix แบบ np.array ที่แต่ละช่องเก็บค่า int)	$H_1 = [1]$ $H_n = \begin{bmatrix} H_{n-1} & H_{n-1} \\ H_{n-1} & -H_{n-1} \end{bmatrix}$ เมื่อ $n > 1$
P(n) เป็นฟังก์ชันที่คืนจำนวนเลข 1 ที่ปรากฏอยู่ใน Hadamard matrix ลำดับที่ n	$P_n = \begin{cases} 3P_{n-1} + M_{n-1}, & n > 1 \\ ??? \text{ คิดเอง}, & n = 1 \end{cases}$
M(n) เป็นฟังก์ชันที่คืนจำนวนเลข -1 ที่ปรากฏอยู่ใน Hadamard matrix ลำดับที่ n	$M_n = \begin{cases} 3M_{n-1} + P_{n-1}, & n > 1 \\ ??? \text{ คิดเอง}, & n = 1 \end{cases}$
S(n) เป็นฟังก์ชันที่คืนผลรวมของตัวเลขทุกตัวที่ปรากฏอยู่ใน Hadamard matrix ลำดับที่ n	$S_n = ???$ คิดเอง ใช้ฟังก์ชันด้านบนให้เป็นประโยชน์

ให้นักเติมโปรแกรมด้านล่างนี้ให้สมบูรณ์

```
import numpy as np

def H(n):
    ???

def P(n):
    ???

def M(n):
    ???

def S(n):
    ???

exec(input().strip()) # do not remove this line
```

ข้อมูลนำเข้า

มีบรรทัดเดียว เป็นคำสั่งที่ใช้ในการทดสอบฟังก์ชัน (ระวัง การใส่ n ที่มีค่ามากเกินไป อาจทำให้โปรแกรมค้างได้)

ข้อมูลส่งออก

แสดงผลการเรียกใช้ฟังก์ชันตามที่กำหนด

ตัวอย่าง

input	output
<code>print(H(3))</code>	<pre>[[1 1 1 1] [1 -1 1 -1] [1 1 -1 -1] [1 -1 -1 1]]</pre>
<code>print(type(H(10)))</code>	<code><class 'numpy.ndarray'></code>
<code>print(P(2))</code>	3
<code>print(M(3))</code>	6
<code>print(S(2))</code>	2