

```
//37 sum of 3D array
```

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
    int con, counter, arr3d[3][3][3], *p = NULL, i, j, k, sum;
```

```
    con = 1;
```

```
    while(con == 1)
```

```
    {
```

```
        p = &arr3d[0][0][0];
```

```
        //ให้ pointer p ใช้ไปที่ array arr3d ช่องที่ 0 0 0 ที่เป็นจุดเริ่ม
```

```
        counter = 0;
```

```
        sum = 0;
```

```
        for(i = 0 ; i < 3 ; i++)
```

```
        {
```

```
            for(j = 0 ; j < 3 ; j++)
```

```
            {
```

```
                for(k = 0 ; k < 3 ; k++)
```

```
                {
```

```
                    printf("input array 3D [%d][%d][%d] : ", i, j, k);
```

```
                    scanf("%d", &arr3d[i][j][k]);
```

```
                    //ให้ใส่ค่าใน array 3 มิติช่องนั้นๆ
```

```
                    sum += *(p+counter);
```

```
                    //ผลรวมเท่ากับค่าใน address ของ p + counter
```

```
                    /*
```

```
                    อธิบาย counter ที่ใช้เป็น counter เนื่องจากว่าการใช้ (p + i + j + k) นั้นจะตรงในช่วงแรก
```

เท่านั้น คือช่วงที่ i, j เป็น 0 และ k มีค่าตั้งแต่ 0 - 2 เนื่องจาก

```
EX
```

ช่องที่	ค่า
---------	-----

0	1
---	---

1	2
---	---

2	3
---	---

3	4
---	---

การใช้ (p + i + j + k)

แล้ว i, j = 0 และ k = 0 จะได้ช่องที่ 0 ซึ่งตรงตามต้องการ

i, j = 0 และ k = 1 จะได้ช่องที่ 1 ซึ่งตรงตามต้องการ

i, j = 0 และ k = 2 จะได้ช่องที่ 2 ซึ่งตรงตามต้องการ

แต่ถ้าเป็น i = 0, j = 1 และ k = 0 จะได้ช่องที่ 1 ที่มีค่าเป็น 0 แต่สิ่งที่ต้องการคือช่องที่ 3 ที่มีค่า

เป็น 4 จึงใช้ตัวแปร counter มาเพิ่มค่าทีละ 1 แทนการใช้ i + j + k

```
*/
```

```
}
```

```
}
```

```
}
```

```
    printf("summation of 3D array is : %d \n", sum);
```

```
    //แสดงผล ผลรวมของ array 3 มิติ
```

```
printf("continues ? (1 for continues) : ");  
scanf("%d", &con);  
//เป็นการถามว่าจะทำต่อไหม และกด 1 เพื่อทำอีกครั้ง
```

```
}
```

```
}
```

Result

```
input array 3D [0][0][0] : 1  
input array 3D [0][0][1] : 1  
input array 3D [0][0][2] : 1  
input array 3D [0][1][0] : 1  
input array 3D [0][1][1] : 1  
input array 3D [0][1][2] : 1  
input array 3D [0][2][0] : 1  
input array 3D [0][2][1] : 1  
input array 3D [0][2][2] : 1  
input array 3D [1][0][0] : 1  
input array 3D [1][0][1] : 1  
input array 3D [1][0][2] : 1  
input array 3D [1][1][0] : 1  
input array 3D [1][1][1] : 1  
input array 3D [1][1][2] : 1  
input array 3D [1][2][0] : 1  
input array 3D [1][2][1] : 1  
input array 3D [1][2][2] : 1  
input array 3D [2][0][0] : 1  
input array 3D [2][0][1] : 1  
input array 3D [2][0][2] : 1  
input array 3D [2][1][0] : 1  
input array 3D [2][1][1] : 1  
input array 3D [2][1][2] : 1  
input array 3D [2][2][0] : 1  
input array 3D [2][2][1] : 1  
input array 3D [2][2][2] : 1  
summation of 3D array is : 27  
continues ? (1 for continues) : |
```