

## Multiplexer

จงเขียนฟังก์ชัน **f1**, **f2**, **f3**, **f4** และ **main** ให้ทำงานตามที่อธิบายกำกับแต่ละฟังก์ชัน ในโครงของโปรแกรมข้างล่างนี้ โดยที่ **a**, **b** และ **c** ที่เป็นพารามิเตอร์ของฟังก์ชัน **f1**, **f2**, **f3** และ **f4** มีลักษณะดังนี้แน่ ๆ ไม่ต้องตรวจอะไรในฟังก์ชัน

- เป็นจำนวนเต็ม มีค่าได้ตั้งแต่ -1000 ถึง 1000
- มีอย่างน้อยค่าหนึ่งใน **a**, **b** หรือ **c** ที่มีค่ามากกว่า 0
- มีอย่างน้อยค่าหนึ่งใน **a**, **b** หรือ **c** ที่มีค่าน้อยกว่า 0

```
def f1(a, b, c):
    # a, b และ c มีค่าตามข้อกำหนดข้างบน
    # คืนจำนวนเต็มที่มีค่าเท่ากับค่าน้อยที่สุดของ a, b และ c คิดเฉพาะตัวที่มีค่ามากกว่า 0

def f2(a, b, c):
    # a, b และ c มีค่าตามข้อกำหนดข้างบน
    # คืนจำนวนเต็มที่มีค่าเท่ากับค่ามากที่สุดของ a, b และ c คิดเฉพาะตัวที่มีค่าน้อยกว่า 0

def f3(a, b, c):
    # a, b และ c มีค่าตามข้อกำหนดข้างบน
    # คืนจำนวนเต็ม (ไม่มีเครื่องหมาย) ที่เป็นตัวเลขหลักแรก (ซ้ายสุด) ของผลบวกของ a, b และ c

def f4(a, b, c):
    # a, b และ c มีค่าตามข้อกำหนดข้างบน
    # คืนจำนวนเต็ม (ไม่มีเครื่องหมาย) ที่เป็นตัวเลขหลักสุดท้าย (ขวาสุด) ของผลบวกของ a, b และ c

def main():
    # รับข้อมูลจากคีย์บอร์ดเป็นจำนวนเต็ม 5 จำนวน s1 s2 a b c คั่นด้วยช่องว่างในบรรทัดเดียวกัน
    # ค่าของ a, b และ c เป็นไปตามข้อกำหนดข้างบน
    # ถ้า s1 เป็น 0 และ s2 เป็น 0 ให้ print ค่าของ f1(a, b, c)
    # ถ้า s1 เป็น 0 และ s2 เป็น 1 ให้ print ค่าของ f2(a, b, c)
    # ถ้า s1 เป็น 1 และ s2 เป็น 0 ให้ print ค่าของ f3(a, b, c)
    # ถ้า s1 เป็น 1 และ s2 เป็น 1 ให้ print ค่าของ f4(a, b, c)
    # ถ้า s1 หรือ s2 เป็นค่าอื่น ให้แสดงคำว่า Error

exec(input().strip()) # DON'T remove this line
```

## ตัวอย่าง

input (จากแป้นพิมพ์)	output (ทางจอภาพ)
<code>print(f1(2, -18, 4))</code>	2
<code>print(f2(-20, -18, 4))</code>	-18
<code>print(f3(2, -18, 4))</code>	1
<code>print(f4(2, -18, 4))</code>	2
<code>main()</code> 1 1 -2 -18 4	6
<code>main()</code> 10 1 -2 -4 6	Error