



## คอมพิวเตอรื (PC)

คอมพิวเตอรื(PC)เครื่องหนึ่ง มีหน่วยประมวลผลซึ่งทำงานแบบ stack และมี register (หน่วยความจำขนาดเล็กมาก) ที่เก็บตัวเลขได้เพียงหนึ่งตัวเท่านั้น คอมพิวเตอรืดังกล่าวสามารถโปรแกรมด้วยภาษา C-- ซึ่งมีคำสั่งดังนี้

**PUSH <ตัวเลข x>** คือ คำสั่งที่ใช้ใส่ค่าตัวเลข x ไปไว้ด้านบนสุดของ stack [ $-10,000 \leq x \leq 10,000$ ]

**STORE** คือ คำสั่งที่ใช้ย้ายตัวเลขบนสุดของ stack ไปเก็บไว้ใน register

**LOAD** คือ คำสั่งที่ใช้สำเนาตัวเลขใน register ไปเก็บไว้ด้านบนสุดของ stack

**PLUS** คือ คำสั่งที่ใช้ลบสองตัวบนสุดของ stack นำทั้งสองตัวมาบวกกัน และเก็บผลลัพธ์ไว้ด้านบนสุดของ stack

**TIMES** คือ คำสั่งที่ใช้ลบสองตัวบนสุดของ stack แล้วนำทั้งสองตัวมาคูณกัน และเก็บผลลัพธ์ไว้ด้านบนสุดของ stack

**IFZERO <คำสั่งที่ n>** คือ คำสั่งที่ใช้ลบตัวบนสุดของ stack แล้วตรวจสอบว่าเลขดังกล่าวเป็นเลขศูนย์หรือไม่ หากเป็นเลขศูนย์จริงให้คอมพิวเตอรืกระโดดไปทำคำสั่งที่ n [ $0 \leq n \leq 1,000$ ]

**DONE** คือ คำสั่งสิ้นสุดการทำงานของโปรแกรมและพิมพ์ค่าบนสุดของ stack ออกมา

การคำนวณจะคำนวณเรียงตามลำดับ โดยการเริ่มต้นของ stack จะเป็นค่าว่าง และค่าเริ่มต้นใน register คือ 0

**ตัวอย่าง** หากโปรแกรมมี input ดังนี้ (ตัวเลข: แสดงลำดับคำสั่งเท่านั้น มิใช่ส่วนของโปรแกรม)

0: PUSH 5	1: STORE	2: LOAD	3: LOAD	4: PUSH -1
5: PLUS	6: STORE	7: LOAD	8: IFZERO 13	9: LOAD
10: TIMES	11: PUSH 0	12: IFZERO 3	13: DONE	

เมื่อผ่านการคำนวณจนสิ้นสุดคำสั่งที่ 7 ค่าใน stack จะเก็บค่าเป็น *ล่างสุด --> [5,4] <-- บนสุด* และค่าใน register จะมีค่าเป็นเลข 4 คำสั่งที่ 8 จะลบค่าบนสุดทั้งจาก stack แล้วตรวจสอบว่าเป็นเลข 0 หรือไม่ เนื่องจากค่าบนสุดที่ลบทั้งดังกล่าวเป็นเลข 4 ไม่ใช่เลข 0 จึงทำคำสั่งที่ 9 ถัดไปตามปกติ ซึ่งหลังจากสิ้นสุดคำสั่งที่ 11 ค่าใน stack จะเก็บค่าเป็น *ล่างสุด --> [20,0] <-- บนสุด* คำสั่งที่ 12 จะลบค่าบนสุดทั้งจาก stack แล้วตรวจสอบว่าเป็นเลข 0 หรือไม่ เนื่องจากค่าบนสุดเป็นเลข 0 จึงกระโดดย้อนกลับไปทำคำสั่งลำดับที่ 3 (LOAD) เมื่อสิ้นสุดโปรแกรมหาดังกล่าว (โปรแกรมรันมาถึงคำสั่ง DONE ในลำดับคำสั่งที่ 13) โปรแกรมจะพิมพ์ค่า 120 ออกมา



**งานของคุณ** จงเขียนโปรแกรมเพื่อจำลองการทำงานของคอมพิวเตอร์เครื่องดังกล่าว

**ข้อมูลเข้า** บรรทัดแรก ระบุจำนวนคำสั่งของภาษา C-- และบรรทัดที่เหลือ คือ โปรแกรมภาษา C--

**ข้อมูลส่งออก** ผลลัพธ์จากคำสั่งที่เกิดขึ้นจากการทำงานของ PC

ตัวอย่างข้อมูลนำเข้า	ตัวอย่างข้อมูลส่งออก
14  PUSH 5  STORE  LOAD  LOAD  PUSH -1  PLUS  STORE  LOAD  IFZERO 13  LOAD  TIMES  PUSH 0  IFZERO 3  DONE	120

\*คำสั่งสุดท้ายจะจบด้วยคำสั่ง DONE เสมอ



## ข้อกำหนด

หัวข้อ	เงื่อนไข
ข้อมูลนำเข้า	Standard Input (คีย์บอร์ด)
ข้อมูลส่งออก	Standard Output (จอภาพ)
ระยะเวลาสูงสุดที่ใช้ในการประมวลผล ต่อชุดทดสอบหนึ่งชุด	1 วินาที
หน่วยความจำสูงสุดที่ใช้ในการประมวลผล ต่อชุดทดสอบหนึ่งชุด	32 MB
จำนวนชุดทดสอบ (โปรแกรมประมวลผลครั้งละชุดทดสอบ)	10
เงื่อนไขการรับโปรแกรม	โปรแกรมต้องประมวลผลข้อมูลตามตัวอย่างที่ให้มาได้

## ข้อมูลคำสั่งเพิ่มเติม

สำหรับผู้เข้าแข่งขันที่เขียนโปรแกรมด้วยภาษา C ให้ระบุชื่อแฟ้มข้อมูล PC.c และระบุส่วนหัวของโปรแกรกดังนี้

```
/*
```

```
TASK: PC
```

```
LANG: C
```

```
AUTHOR: YourName YourLastName
```

```
CENTER: WU
```

```
*/
```

สำหรับผู้เข้าแข่งขันที่เขียนโปรแกรมด้วยภาษา C++ ให้ระบุชื่อแฟ้มข้อมูล PC.cpp และระบุส่วนหัวของโปรแกรกดังนี้

```
/*
```

```
TASK: PC
```

```
LANG: C++
```

```
AUTHOR: YourName YourLastName
```

```
CENTER: WU
```

```
*/
```