

ใบงานที่ 1

เรื่อง สแตก (Stack)

เสนอ

อาจารย์ ปิยพล ยืนยงสถาวร

จัดทำโดย

นายกฤษฎา วิริยา 65543206041-7

ใบงานนี้เป็นส่วนหนึ่งของรายวิชา โครงสร้างข้อมูลและขั้นตอนวิธี
หลักสูตรวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์
มหาวิทยาลัยเทคโนโลยีราชมงคลล้านนา
ประจำภาคที่ 1 ปีการศึกษา 2566

คำสั่ง/คำชี้แจง

- แสดงโค้ดโปรแกรมเป็นส่วนๆพร้อมทั้งอธิบาย
- แสดงผลการรันโปรแกรม พร้อมอธิบายการทำงาน
- สรุปผลการทดลอง

ลำดับดับขั้นการทดลอง

```
7 #include <stdio.h> //use printf()
8 #include <conio.h> //use getch()
9 #define MaxStack 6 // Set Max Stack
10 int stack[MaxStack]; // Declare Max Stack 0..5
11 int x; // Temperature variable
12 int SP = 0; // Initial SP=0
13 char status = 'N'; // Initial Status = NORMAL
14 char ch; // KBD Read variable
```

- #include <stdio.h>: เป็นส่วนของโค้ดที่ใช้ระบุว่าโปรแกรมต้องการใช้ฟังก์ชันที่เกี่ยวข้องกับการ ป้อนออกข้อมูล
- #include <conio.h>: เป็นส่วนของโค้ดที่ใช้ระบุว่าโปรแกรมต้องการใช้ฟังก์ชันที่เกี่ยวข้องกับการ ควบคุมอินพุตทางคีย์บอร์ด getch()
- #define MaxStack 6: เป็นการกำหนดค่า MaxStack โดยกำหนดค่าให้เป็น 6 ซึ่งแทนจำนวน สูงสุดของข้อมูลที่ Stack สามารถเก็บได้
- int stack[MaxStack]: เป็นการประกาศตัวแปร array ชื่อ stack ที่มีขนาดเท่ากับ MaxStack ซึ่ง ใช้เก็บข้อมูลใน Stack
- int x: เป็นการประกาศตัวแปรชื่อ x ที่ใช้เก็บค่าชั่วคราว
- int SP = 0: เป็นการกำหนดค่าเริ่มต้นของตัวแปร SP (Stack Pointer) ให้เท่ากับ 0
- **char status = 'N':** เป็นการกำหนดค่าเริ่มต้นของตัวแปร status ให้เป็น 'N' ซึ่งเป็นสถานะ NORMAL
- char ch: เป็นการประกาศตัวแปรชื่อ ch ที่ใช้เก็บค่าอินพุตที่รับมาจากคีย์บอร์ด

push(int x): เป็นฟังก์ชันที่ใช้ในการเพิ่มข้อมูลลงใน Stack (PUSH) โดยมีขั้นตอนดังนี้:

- ตรวจสอบว่า Stack เต็มหรือไม่ ถ้าเต็มจะแสดงข้อความ "!!!OVER FLOW!!!..." และกำหนดค่า status เป็น 'O' หมายถึงสถานะ OVFR FLOW
- ถ้า Stack ยังไม่เต็ม จะเพิ่มค่า SP (Stack Pointer) ไป 1 หน่วยเพื่อชี้ไปยังตำแหน่งว่างใน Stack นำ ข้อมูล x ที่ต้องการ PUSH ไปเก็บใน Stack ที่ตำแหน่งที่ SP ชี้ไป

```
26 int pop() { // POP Function
27    int x;
28    if (SP # 0) { // Check Stack NOT EMPTY?
29         x = stack[SP]; // Get data from Stack
30         stack[SP] = 0; // Clear
31         SP--; // Decrease SP
32         return (x); // Return data
33    } else {
        printf("\n!!!UNDER FLOW!!!...\n");
        status = 'U'; // set STATUS = "UNDER FLOW"
36    }
37    return -1; // Return data
38 }
```

pop(): เป็นฟังก์ชันที่ใช้ในการนำข้อมูลออกจาก Stack (POP) โดยมีขั้นตอนดังนี้:

- ตรวจสอบว่า Stack ไม่ว่างหรือไม่ ถ้าว่างจะแสดงข้อความ "!!!UNDER FLOW!!!..." และกำหนดค่า status เป็น 'U' หมายถึงสถานะ UNDER FLOW
- ถ้า Stack ไม่ว่าง จะนำข้อมูลที่อยู่ในตำแหน่ง SP ของ Stack ออกมาเก็บในตัวแปร x
- เคลียร์ค่าในตำแหน่ง SP ของ Stack
- ลดค่า SP ลง 1 หน่วยเพื่อชี้ไปยังตำแหน่งล่างสุดของข้อมูลใน Stack
- ส่งค่า x กลับเป็นผลลัพธ์ของการ POP ข้อมูล

ShowAllStack(): เป็นฟังก์ชันที่ใช้ในการแสดงข้อมูลทั้งหมดที่อยู่ใน Stack โดยมีขั้นตอนดังนี้:

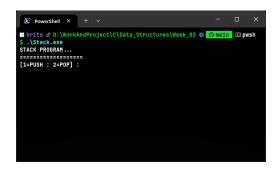
- แสดงค่า N ซึ่งเป็นขนาดของ Stack ตามการกำหนดค่า MaxStack
- แสดงค่า status ซึ่งเป็นสถานะปัจจุบันของ Stack
- แสดงค่า SP ซึ่งเป็นตำแหน่งของ Stack Pointer
- วนลูปแสดงข้อมูลที่อยู่ใน Stack ทั้งหมด โดยเริ่มจากตำแหน่งที่ 1 ถึงตำแหน่ง MaxStack-1

main(): เป็นฟังก์ชันหลักที่เรียกใช้ฟังก์ชันและโค้ดส่วนอื่น ๆ เพื่อเริ่มต้นการทำงานของโปรแกรม

- แสดงข้อความเริ่มต้น "STACK PROGRAM..."
- เริ่มลูป while ทำงานตราบเท่าที่สถานะ (status) เป็น 'N' (NORMAL)
- แสดงเมนูให้ผู้ใช้เลือก 1=PUSH หรือ 2=POP และรอรับอินพุตจากคีย์บอร์ด (KBD)
- ตรวจสอบค่าที่รับเข้ามา (ch) และดำเนินการตามเงื่อนไข:

- ถ้าผู้ใช้เลือก 1 (PUSH) จะแสดงข้อความ "Enter Number : " เพื่อรับข้อมูลจากคีย์บอร์ด (x) และ เรียกใช้ฟังก์ชัน push(x) เพื่อ PUSH ข้อมูลลงใน Stack และแสดงข้อมูลทั้งหมดใน Stack โดย เรียกใช้ฟังก์ชัน ShowAllStack()
- ถ้าผู้ใช้เลือก 2 (POP) จะเรียกใช้ฟังก์ชัน pop() เพื่อ POP ข้อมูลจาก Stack และแสดงข้อมูลที่ POP ออกมา และแสดงข้อมูลทั้งหมดใน Stack โดยเรียกใช้ฟังก์ชัน ShowAllStack()

Run Program



เริ่มโปรแกรมจะมีแสดงข้อความและจะมีการให้ Input ค่า Menu ลงไป โดย 1 คือเพิ่มข้อมูล 2 คือลบข้อมูล

PUSH เพิ่มข้อมูล จะเพิ่มข้อมูลเก็บไว้ ถ้าข้อมูลเต็มแล้วจะแสดง !!!OVER FLOW!!!... และ Status เป็น O

POP ลบข้อมูล จะลบข้อมูลที่รับมาล่าสุดก่อน ถ้าข้อมูลที่ลบไม่มีแล้วจะแสดง !!!UNDER FLOW!!!... และ Status เป็น U

สรุปผลการทดลอง

Stack หรือชุดข้อมูลแบบ First-In-Last-Out (FILO) โดยมีความสามารถในการ PUSH (เพิ่มข้อมูลลงใน Stack) และ POP (นำข้อมูลออกจาก Stack) โดยผู้ใช้สามารถทำการ PUSH หรือ POP ข้อมูลได้ตามต้องการ โดย โปรแกรมจะแสดงสถานะและข้อมูลที่อยู่ใน Stack หลังจากทำการ PUSH หรือ POP ข้อมูลแต่ละครั้ง