

ใบงานปฏิบัติการ เรื่อง Hash Table ครั้งที่ 2

จงเขียนโปรแกรมด้วยภาษา Python เพื่อรับค่าขนาดของตารางแฮช (Hash table) ที่ผู้ใช้ป้อนทางแป้นพิมพ์ แล้วสร้างตารางแฮชที่มีขนาดดังกล่าว จากนั้นวนซ้ำรับค่าคีย์ (key) ที่มีค่ามากกว่าหรือเท่ากับ 0 และข้อมูลชื่อสินค้าเพื่อจัดเก็บในตารางแฮชโดยใช้วิธี Open Addressing แบบ Linear Probing ในการแก้ปัญหาการชนกันแล้วแสดงที่อยู่ (Address) ทางจอภาพโดยโปรแกรมจะหยุดการวนซ้ำเมื่อคีย์มีค่าน้อยกว่า 0 หลังจากหยุดการวนซ้ำโปรแกรมจะแสดงที่อยู่ (Address) และข้อมูลที่จัดเก็บในตารางแฮช ถ้าไม่มีข้อมูลที่จัดเก็บในที่อยู่นั้นให้แสดงข้อความ “ไม่มีข้อมูลจัดเก็บ” ทางจอภาพ

ตัวอย่างผลลัพธ์:

1. เริ่มการทำงานของโปรแกรม

โปรดป้อนขนาดตารางแฮช:

2. ตัวอย่างการวนซ้ำรับข้อมูลเพื่อจัดเก็บในตารางแฮช

โปรดป้อนคีย์ (key) ที่มีค่ามากกว่าหรือเท่ากับ 0: 20

โปรดป้อนข้อมูลชื่อสินค้าเพื่อจัดเก็บในตารางแฮช: หนังสือ

Address = 9

โปรดป้อนคีย์ (key) ที่มีค่ามากกว่าหรือเท่ากับ 0: 5

โปรดป้อนข้อมูลชื่อสินค้าเพื่อจัดเก็บในตารางแฮช: ปากกา

Address = 5

โปรดป้อนคีย์ (key) ที่มีค่ามากกว่าหรือเท่ากับ 0: 9

โปรดป้อนข้อมูลชื่อสินค้าเพื่อจัดเก็บในตารางแฮช: เครื่องพิมพ์

Address = 10

โปรดป้อนคีย์ (key) ที่มีค่ามากกว่าหรือเท่ากับ 0: 16

โปรดป้อนข้อมูลชื่อสินค้าเพื่อจัดเก็บในตารางแฮช: กระดาษ A4

Address = 6

โปรดป้อนคีย์ (key) ที่มีค่ามากกว่าหรือเท่ากับ 0: 29

โปรดป้อนข้อมูลชื่อสินค้าเพื่อจัดเก็บในตารางแฮช: ลำโพง

Address = 7

โปรดป้อนคีย์ (key) ที่มีค่ามากกว่าหรือเท่ากับ 0: -2

โปรดป้อนข้อมูลชื่อสินค้าเพื่อจัดเก็บในตารางแฮช: ไม้ไคร้

3. ตัวอย่างการแสดงผลข้อมูลที่จัดเก็บในตารางแฮช

ข้อมูลที่จัดเก็บในตารางแฮช

Address = 0 ว่างไม่มีข้อมูลจัดเก็บ

Address = 1 ว่างไม่มีข้อมูลจัดเก็บ

Address = 2 ว่างไม่มีข้อมูลจัดเก็บ

Address = 3 ว่างไม่มีข้อมูลจัดเก็บ

Address = 4 ว่างไม่มีข้อมูลจัดเก็บ

Address = 5 ข้อมูลที่จัดเก็บคือ ปากกา

Address = 6 ข้อมูลที่จัดเก็บคือ กระดาษ A4

Address = 7 ข้อมูลที่จัดเก็บคือ ลำโพง

Address = 8 ว่างไม่มีข้อมูลจัดเก็บ

Address = 9 ข้อมูลที่จัดเก็บคือ หนังสือ

Address = 10 ข้อมูลที่จัดเก็บคือ เครื่องพิมพ์