**01076104 Programming Project**

**ภาควิชาวิศวกรรมคอมพิวเตอร์ คณะวิศวกรรมศาสตร์**

**สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง**

**การทดลองที่ 4 : การทดสอบโปรแกรมและการ Debug**

**การทดลองที่ 4 : การทดสอบโปรแกรมและการ Debug**

**จุดประสงค์**

นักศึกษาสามารถทดสอบโปรแกรมเพื่อค้นหาความผิดพลาดในโปรแกรม และ Debug เพื่อแก้ความผิดพลาดในโปรแกรมที่เขียนได้

**ตอนที่ 1 : ให้นักศึกษาเขียนโปรแกรมที่กำหนด**

1. ให้นักศึกษาเขียนโปรแกรมรับ string 1 ชุด แล้วแสดงผลชุดตัวอักษร โดยถ้ามีตัวเลขหน้าตัวอักษรใด ให้แสดงผลตัวอักษรนั้นซ้ำเท่ากับจำนวนตัวเลข เช่น

Input : a2Bc4d11Eg0Nk

Output : aBBcddddEEEEEEEEEEEgk

## Source Code

|  |  |
| --- | --- |
| บรรทัด | โปรแกรม |
| 1  2  3  4  5  6  7  8  9  10  11  12  13  14  15  16  17  18 |  |

## การทำงานของโปรแกรม

|  |  |  |
| --- | --- | --- |
| บรรทัดที่ | ถึงบรรทัดที่ | การทำงาน |
| 1 | 1 | ประกาศการใช้ stdio.h เพื่อใช้สำหรับการนำเข้า/ส่งออกข้อมูล |
| 4 | 4 | ประกาศ str สำหรับเก็บข้อมูลนำเข้า |
| 5 | 5 | รับข้อมูลเก็บไว้ใน str |
| 7 | 12 | ประมวลผล str เป็นผลลัพธ์ตามที่โจทย์ต้องการ |
| 7 | 7 | กำหนดให้ i = 0  จบการทำงานเมื่อถึงจุดสิ้นสุดของ str  เพิ่มค่า i เป็นจำนวน 1 เมื่อเริ่มรอบถัดไป |
| 8 | 8 | ถ้า str[i] เป็นตัวเลข  n = 0.  นอกเหนือจากนั้น  n = 1. |
| 10 | 11 | ทำซ้ำถ้า str[i] เป็นตัวเลข  n += (n \* 10) + (str[i] – ‘0’).  (แปลงส่วนของ str ที่เป็นตัวเลขไว้ในตัวแปร n โดย) |
| 13 | 14 | ทำซ้ำถ้า n != 0  แสดงตัวอักษรที่ i ใน str |
| 17 | 17 | สิ้นสุดโปรแกรม |

# ตอนที่ 2 ให้นักศึกษากำหนด Test Script เพื่อทดสอบการทำงานของโปรแกรม

|  |  |  |  |
| --- | --- | --- | --- |
| ลำดับที่ | Input ที่กำหนด | Output ที่คาดหวัง | ผลทดสอบ (🗸/🗴) |
| 1 | a2Bc4d11Eg0Nk | aBBcddddEEEEEEEEEEEgk | 🗸 |
| 2 | 0a | (blank) | 🗸 |
| 3 | 12 | (blank) | 🗴 |
| 4 | 0012a | aaaaaaaaaaaa | 🗸 |
| 5 | abc | abc | 🗸 |
| 6 | a9 | a | 🗴 |
| 7 | 0 | (blank) | 🗴 |
| 8 | a1c | ac | 🗸 |
| 9 | aaa | aaa | 🗸 |
| 10 | 0000a | (blank) | 🗸 |
| 11 | 11a | aaaaaaaaaaa | 🗸 |
| 12 | ggwp | ggwp | 🗸 |

# ตอนที่ 3 ให้นักศึกษาระบุค่าของตัวแปรที่ตำแหน่งสำคัญเพื่อเป็นข้อมูลสำหรับการ Debug โปรแกรม

|  |  |
| --- | --- |
| บรรทัดที่กำหนด Breakpoint | ตัวแปรสำคัญที่จะ Watch ดูข้อมูล |
| 9 | n, i, str[i] |
| 12 | n, i, str[i] |
| 15 | n, i, str[i] |
| 8 | i, str[i] |

การกําหนดค่าตัวแปรในแต่ละ Break Point ของ Test Script แต่ละข้อ พร้อมผลการทดสอบ

(ให้นักศึกษาเลือกทดสอบ Test Script อย่างน้อย 3 อัน ที่ผลการทดสอบอาจเป็น 🗴)

|  |  |  |  |
| --- | --- | --- | --- |
| Test Script ที่ | บรรทัดที่กำหนด Breakpoint | ค่าตัวแปรที่ควรเป็น | ผลทดสอบ (🗸/🗴) |
| 3 | 9 รอบที่ 1 | n = 0, i = 0, str[i] = ‘1’ | 🗸 |
| 12 รอบที่ 1 | n = 12, i = 2, str[i] = ‘\0’ | 🗸 |
| 15 รอบที่ 1 | n = 12, i = 2, str[i] = ‘\0’ | 🗸 |
| 8 | i = 2, str[i] = ‘\0’ (จบการวนซ้ำ) | 🗴 |
| 6 | 9 รอบที่ 1 | n = 1, i = 0, str[i] = ‘a’ | 🗸 |
| 12 รอบที่ 1 | n = 1, i = 0, str[i] = ‘a’ | 🗸 |
| 15 รอบที่ 1 | n = 1, i = 0, str[i] = ‘a’ | 🗸 |
| 9 รอบที่ 2 | n = 1, i = 1, str[i] = ‘9’ | 🗸 |
| 12 รอบที่ 2 | n = 9, i = 2, str[i] = ‘\0’ | 🗸 |
| 15 รอบที่ 2 | n = 9, i = 2, str[i] = ‘\0’ | 🗸 |
| 8 | i = 2, str[i] = ‘\0’ (จบการวนซ้ำ) | 🗴 |
| 7 | 9 รอบที่ 1 | n = 0, i = 0, str[i] = ‘0’ | 🗸 |
| 12 รอบที่ 1 | n = 0, i = 1, str[i] = ‘\0’ | 🗸 |
| 15 รอบที่ 1 | n = 12, i = 1, str[i] = ‘\0’ | 🗸 |
| 8 | i = 1, str[i] = ‘\0’ (จบการวนซ้ำ) | 🗴 |

ในกรณีที่ผลทดสอบเป็น 🗴 ให้นักศึกษาระบุว่า เหตุใดจึงเป็นเช่นนั้น และจะแก้ไขโปรแกรมอย่างไร

|  |
| --- |
| **Test Script ที่ 3, 6, และ 7**  **ข้อผิดพลาด** โปรแกรมจะทำการพิมพ์ตัวอักษรต่อไปแม้ว่าจะสิ้นสุด str แล้วก็ตาม ในกรณีที่ str นั้นสิ้นสุดด้วยตัวเลข  **สาเหตุ**  หลังจากที่โปรแกรมพิมพ์ ‘\0’ ภายในลูปจะทำการข้ามไปตัวถัดไป จึงทำให้การอ่านค่า str เกิด out of bound  **วิธีแก้ไข**  เปลี่ยนวิธีการตรวจสอบการสิ้นสุดของ str โดยการตรวจสอบตามความยาวแทนการใช้ null terminator character |