

Lab#7 – White-box testing

วัตถุประสงค์การเรียนรู้

1. ผู้เรียนสามารถออกแบบการทดสอบแบบ White-box testing ได้
2. ผู้เรียนสามารถวิเคราะห์ปัญหาด้วย Control flow graph ได้
3. ผู้เรียนสามารถออกแบบกรณีทดสอบโดยคำนึงถึง Line coverage ได้
4. ผู้เรียนสามารถออกแบบกรณีทดสอบโดยคำนึงถึง Block coverage ได้
5. ผู้เรียนสามารถออกแบบกรณีทดสอบโดยคำนึงถึง Branch coverage ได้
6. ผู้เรียนสามารถออกแบบกรณีทดสอบโดยคำนึงถึง Condition coverage ได้
7. ผู้เรียนสามารถออกแบบกรณีทดสอบโดยคำนึงถึง Branch and Condition coverage ได้

โจทย์: CLUMP COUNTS

Clump counts (<https://codingbat.com/prob/p193817>) เป็นโปรแกรมที่ใช้ในการนับการเกาะกลุ่มกันของข้อมูลภายใน Array โดยการเกาะกลุ่มกันจะนับสมาชิกใน Array ที่อยู่ติดกันและมีค่าเดียวกันตั้งแต่สองตัวขึ้นไปเป็นหนึ่งกลุ่ม เช่น

[1, 2, 2, 3, 4, 4] → 2

[1, 1, 2, 1, 1] → 2

[1, 1, 1, 1, 1] → 1

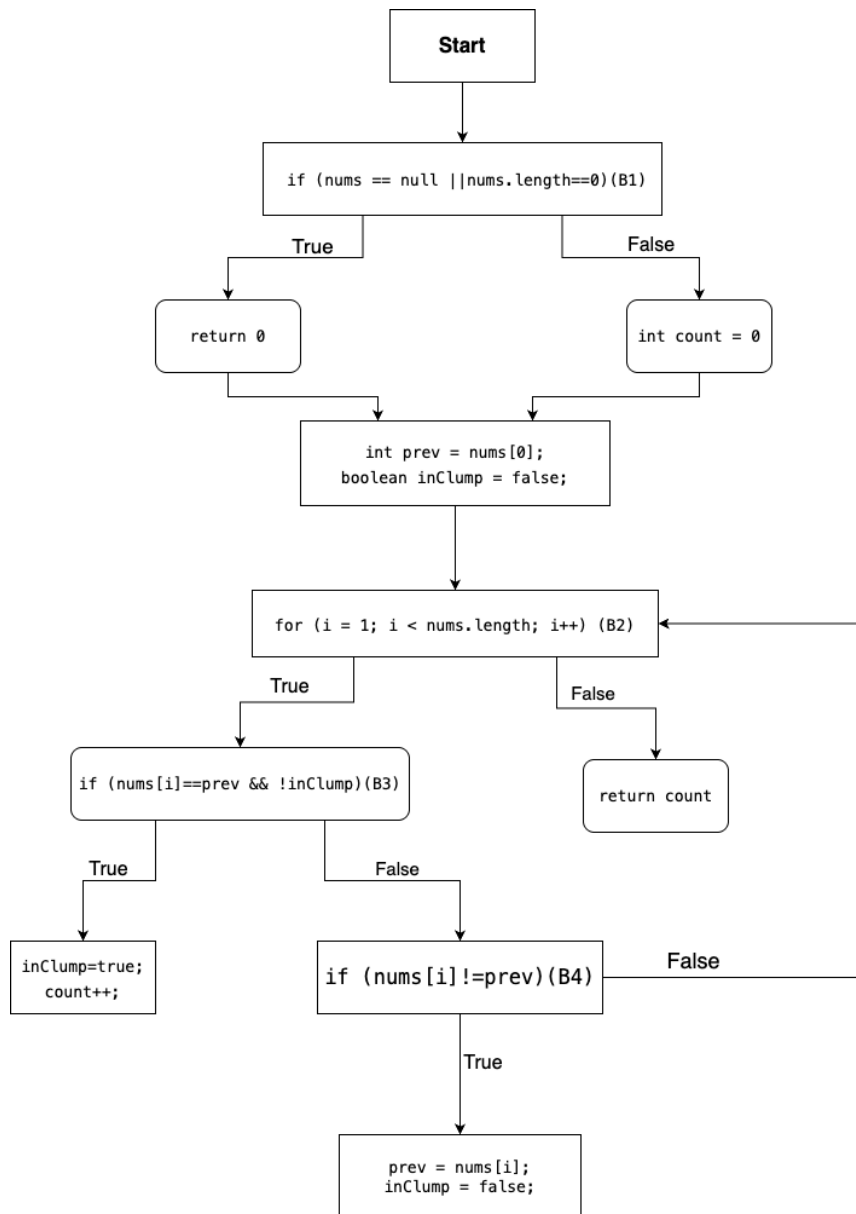
ซอร์สโค้ดที่เขียนขึ้นเพื่อนับจำนวนกลุ่มของข้อมูลที่อยู่ติดกันอยู่ที่

<https://github.com/ChitsuthaCSKKU/SQA/tree/2025/Assignment/Lab7> โดยที่ nums เป็น Array ที่ใช้ในการสนับสนุนการนับกลุ่มของข้อมูล (Clump) ทำให้ nums เป็น Array ที่จะต้องไม่มีค่าเป็น Null และมีความยาวมากกว่า 0 เสมอ หาก nums ไม่เป็นไปตามเงื่อนไขที่กำหนดนี้ โปรแกรมจะ return ค่า 0 แทนการ return จำนวนกลุ่มของข้อมูล

แบบฝึกปฏิบัติที่ 7.1 CONTROL FLOW GRAPH

จากโจทย์และ Source code ที่กำหนดให้ (CountWordClumps.java) ให้เขียน Control Flow Graph (CFG) ของเมธอด countClumps() จากนั้นให้ระบุ Branch และ Condition ทั้งหมดที่พบใน CFG ให้ครบถ้วน

ตอบ



Branch:

1. จาก if (nums == null || nums.length == 0)
 - true: กลับออก return 0
 - false: ไปทำงานต่อ
2. for-loop (i < nums.length)

Lab instruction

- true: เข้า loop (อาจเข้าหลายรอบ)
- false: ออกจาก loop
- 3. if (nums[i] == prev && !inClump)
 - true: เข้าเพิ่ม clump
 - false: ไปต่อ (ไม่เพิ่ม clump)
- 4. if (nums[i] != prev)
 - true: อัปเดต prev/inClump
 - false: ไม่ทำอะไร

Condition:

- C1: (nums == null)
 - C2: (nums.length == 0) (ทั้งสองนี้อยู่ใน if เดียวกัน เชื่อม OR)
 - C3: (nums[i] == prev)
 - C4: (!inClump)
 - (C3 && C4) รวมใน if ที่ตรวจ clump ใหม่
 - C5: (nums[i] != prev)
-

แบบฝึกปฏิบัติที่ 7.2 LINE COVERAGE

1. จาก Control Flow Graph (CFG) ของเมธอด countClumps() ในข้อที่ 1 ให้ออกแบบกรณีทดสอบเพื่อให้ได้ Line coverage = 100%
2. เขียนกรณีทดสอบที่ได้ พร้อมระบุบรรทัดที่ถูกตรวจสอบทั้งหมด
3. แสดงวิธีการคำนวณค่า Line coverage

CP353201 Software Quality Assurance (1/2568)

ผศ.ดร.ชิตสุธา สุ่มเล็ก

Lab instructionตอบ

Test Case No.	Input(s)	Expected Result(s)	Path and Branch
TC1	null	0	1, 2
TC2	[]	0	1, 2
TC3	[1]	0	1, 3, 4, 5, 6, 13
TC4	[1,2,2,3]	1	1, 3, 4, 5, 6-13 ทุกบรรทัด

Line coverage = $13 / 13 * 100 = 100\%$

แบบฝึกปฏิบัติที่ 7.3 BLOCK COVERAGE

1. จาก Control Flow Graph (CFG) ของเมธอด countClumps() ในข้อที่ 1 ให้ออกแบบกรณีทดสอบเพื่อให้ได้ Block coverage = 100%
2. เขียนกรณีทดสอบที่ได้ พร้อมระบุ Block ที่ถูกตรวจสอบทั้งหมด
3. แสดงวิธีการคำนวณค่า Block coverage

ตอบ

CP353201 Software Quality Assurance (1/2568)

ผศ.ดร.ชิตสุธา สุ่มเล็ก

Lab instruction

```
public static int countClumps(int[] nums) {  
    if (nums == null || nums.length == 0) {           // Block A  
        return 0;                                     // Block B  
    }  
    int count = 0;                                     // Block C  
    int prev = nums[0];  
    boolean inClump = false;  
    for (int i = 1; i < nums.length; i++) {           // Block D (วนลูป)  
        if (nums[i] == prev && !inClump) {           // Block E (เริ่มต้น clump ใหม่)  
            inClump = true;                           // Block F (action ใน clump)  
            count++;  
        }  
        if (nums[i] != prev) {                         // Block G (เปลี่ยนกลุ่ม)  
            prev = nums[i];                             // Block H (action เปลี่ยนกลุ่ม)  
            inClump = false;  
        }  
    }  
    return count;                                     // Block I  
}
```

Test Case No.	Input(s)	Expected Result(s)	Path and Branch
TC1	null	0	Block A (true) → Block B
TC2	[]	0	Block A (false,false) → Block B

CP353201 Software Quality Assurance (1/2568)

ผศ.ดร.ชิตสุธา สุ่มเล็ก

Lab instruction

TC3	[1]	0	Block A (false), C, D (loop ไม่เข้า), I
TC4	[1,2,2,3]	1	A(false), C, D(loop), E/F (เจอ clump), G/H (เปลี่ยนกลุ่ม), I
TC5	[2,2,2,2]	1	A(false), C, D(loop), E/F (ยาว), G (false, ไม่เปลี่ยน), I

Block coverage = $9 / 9 \times 100 = 100\%$

แบบฝึกปฏิบัติที่ 7.3 BRANCH COVERAGE

- จาก Control Flow Graph (CFG) ของเมธอด countClumps() ในข้อที่ 1 ให้ออกแบบกรณีทดสอบเพื่อให้ได้ Branch coverage = 100%
- เขียนกรณีทดสอบที่ได้ พร้อมระบุ Path และ Branch ที่ถูกตรวจสอบทั้งหมด
- แสดงวิธีการคำนวณค่า Branch coverage

ตอบ

Test Case No.	Input(s)	Expected Result(s)	Path	Branch
---------------	----------	--------------------	------	--------

CP353201 Software Quality Assurance (1/2568)

ผศ.ดร.ชิตสุธา สุ่มเล็ก

Lab instruction

TC1	null	0	if (nums == null) → true → return 0	1-true
TC2	[]	0	if (nums == null) → false, (nums.length == 0) → true → return 0	1-true
TC3	[1]	0	if (..)→false, ไม่เข้า for → return	1-false, for: ไม่เข้า (loop exit)
TC4	[1,2,3]	0	if (..)→false, for (i=1,2), (nums[i]==prev && !inClump)→false, (nums[i]!=prev)→true	2-false, 3-true
TC5	[1,1,2]	1	for-loop, (nums[i]==prev && !inClump)→true, (nums[i]!=prev)→true	2-true, 3-true

CP353201 Software Quality Assurance (1/2568)

ผศ.ดร.ชิตสุธา สุ่มเล็ก

Lab instruction

TC6	[2,2,2,2]	1	for-loop, (nums[i]==prev && !inClump)→true/false, (nums[i]!=prev)→false	2-true, 2-false, 3-false
-----	-----------	---	--	--------------------------

Branch coverage == $6 / 6 \times 100 = 100\%$

แบบฝึกปฏิบัติที่ 7.4 CONDITION COVERAGE

1. จาก Control Flow Graph (CFG) ของเมธอด countClumps() ในข้อที่ 1 ให้ออกแบบกรณีทดสอบเพื่อให้ได้ Condition coverage = 100%
2. เขียนกรณีทดสอบที่ได้ พร้อมระบุ Path และ Condition ที่ถูกตรวจสอบทั้งหมด เช่น Condition A = T และ Condition B = F
3. แสดงวิธีการคำนวณค่า Condition coverage

ตอบ

Test Case No.	Input(s)	Expected Result(s)	Path and Condition
---------------	----------	--------------------	--------------------

CP353201 Software Quality Assurance (1/2568)

ผศ.ดร.ชิตสุธา สุ่มเล็ก

Lab instruction

TC1	null	0	A=T, (B ไม่ตรวจ), C, D ถูก bypass
TC2	[]	0	A=F, B=T, C, D ถูก bypass
TC3	[1]	0	A=F, B=F, (for ไม่เข้า)
TC4	[1,2,2]	1	A=F, B=F, i=1: C=F, D=ไม่เข้า; i=2: C=T, D=T
TC5	[2,2,2,2]	1	A=F, B=F, i=1: C=T, D=T, i=2: C=T, D=F, i=3: C=T, D=F
TC6	[1,1,2]	1	A=F, B=F, i=1: C=T, D=T; i=2: C=F, D=ไม่เข้า

CP353201 Software Quality Assurance (1/2568)

ผศ.ดร.ชิตสุธา สุ่มเล็ก

Lab instruction

TC7	[1,2,1]	0	A=F, B=F, i=1: C=F, D=ไม่เข้า; i=2: C=F, D=ไม่เข้า
-----	---------	---	---

Condition coverage = $(8 / 8) \times 100 = 100\%$

แบบฝึกปฏิบัติที่ 7.5 BRANCH AND CONDITION COVERAGE (C/DC COVERAGE)

1. จาก Control Flow Graph (CFG) ของเมธอด countClumps() ในข้อที่ 1 ให้ออกแบบกรณีทดสอบให้ได้ C/DC coverage = 100%
2. เขียนกรณีทดสอบที่ได้ พร้อมระบุ Path, Branch, และ Condition ที่ถูกตรวจสอบทั้งหมด
3. แสดงวิธีการคำนวณค่า C/DC coverage
4. เขียนโค้ดสำหรับทดสอบตามกรณีทดสอบที่ออกแบบไว้ด้วย JUnit และบันทึกผลการทดสอบ

ตอบ

Test Case No.	Input(s)	Expected Result(s)	Actual Result(s)	Path, Branch, and Condition	Pass/Fail
---------------	----------	--------------------	------------------	-----------------------------	-----------

CP353201 Software Quality Assurance (1/2568)

ผศ.ดร.ชิตสุธา สุ่มเล็ก

Lab instruction

TC1	null	0	0	if (nums == null): T; รับ return ตรงนี้ (A=T) Branch 1: T	Pass
TC2	new int[]{}	0	0	if (nums == null): F; if (nums.length==0): T (B=T) Branch 1: T	Pass
TC3	new int[]{5}	0	0	if (nums == null): F; if (nums.length==0): F; ไม่มี loop	Pass

CP353201 Software Quality Assurance (1/2568)

ผศ.ดร.ชิตสุธา สุ่มเล็ก

Lab instruction

TC4	new int[]{1,2, 3,4,5}	0	0	if (nums == null): F; if (nums.length==0): F; for loop ผ่าน หมด C=F ทุกครั้ง (Branch 3 F); never enters D	Pass
TC5	new int[]{6,6, 6,6,6}	1	1	เข้า for, C=T(ครั้ง แรก), D=T (นับ clump); ต่อไป C=T, D=F หลาย ครั้ง (inClump true อยู่ ไม่เพิ่ม ซ้ำ), ครบทุก Branch4 T/F	Pass

TC6	new int[] {1,2, 2,3}	1	1	มี C=F (i=1:1!=2), C=T,D=T (i=2:2==2,inClump=false), C=F (i=3:3!=2),ครอบ คลุมทางมี clump เดียวกลาง	Pass
TC7	new int[] {7,7, 8,9,10,1 0}	2	2	C=T,D=T (i=1), init clump, C=F หลายครั้ง, C=T,D=T (i=5), test Branch3/4 สองรอบ	Pass

CP353201 Software Quality Assurance (1/2568)

ผศ.ดร.ชิตสุธา สุ่มเล็ก

Lab instruction

TC8	new int[] {1,1, 2,2,3,3,3 ,4}	3	3	หลาย clump, C=T,D=T (3 ครั้ง), C=F สลับหลาย กลุ่ม,ครอบคลุม หลาย Branch3/4	Pass
TC9	new int[] {1,1, 2,1,1}	2	2	clump หน้า, C=T,D=T (i=1), ตรงกลาง/ท้าย C=T,D=T (i=4), D=F ระหว่างเมื่อ ยัง inClump	Pass
TC10	new int[] {4,4, 4,4}	1	1	C=T, D=T(i=1), D=F(i=2,3), สำรวจกรณีเดียว กับ long clump	Pass
TC11	new int[] {1,2, 1,2,1}	0	0	C=F ทุกครั้ง (Never triggers D); Loop ไม่เข้า clump	Pass

C/DC coverage = 11/11 *100 =100%