

HW #5: Due June 6th 23:59

1. Generate test cases that covers all possible execution paths of the triangle program through depth first search (DFS) traversal.
 - ① Assume that initial test case is (1,1,1)
 - ② Write down a current executed symbolic path condition ϕ and a next symbolic path condition ψ obtained through DFS traversal
 - ③ Write down a LIA SMT formula for the next symbolic path condition ψ
 - ④ Solve the LIA SMT specification through Z3
 - ① If UNSAT, try to negate another branch in a DFS order
 - ② If SAT, record the solution as next input and repeat from Step 2 until all paths are covered/

You have to submit

- a completed table in page 4 that contains test cases and all executed symbolic path conditions and next PCs
- a completed execution tree whose leaves are marked with TCs or UNSAT in pg 4
- All LIA SMT specifications for ψ s and their solutions by Z3 (submit both hardcopy and softcopy that should be sent to hongshin@gmail.com)

```
#include <crest.h>
```

```
int main() {
```

```
    int a,b,c, match=0;
```

```
    CREST_int(a); CREST_int(b); CREST_int(c);
```

```
    // filtering out invalid inputs
```

```
    if(a <= 0 ) exit(); if(b <= 0 ) exit(); if(c <= 0 ) exit();
```

```
    printf("a,b,c = %d,%d,%d:",a,b,c);
```

```
    //0: Equilateral, 1:Isosceles,
```

```
    // 2: Not a triangle, 3:Scalene
```

```
    int result=-1;
```

```
    if(a==b) match=match+1;
```

```
    if(a==c) match=match+2;
```

```
    if(b==c) match=match+3;
```

```
    if(match==0) {
```

```
        if( a+b <= c) result=2;
```

```
        else if( b+c <= a) result=2;
```

```
        else if(a+c <= b) result =2;
```

```
        else result=3;
```

```
    } else {
```

```
        if(match == 1) {
```

```
            if(a+b <= c) result =2;
```

```
            else result=1;
```

```
        } else {
```

```
            if(match ==2) {
```

```
                if(a+c <=b) result = 2;
```

```
                else result=1;
```

```
            } else {
```

```
                if(match==3) {
```

```
                    if(b+c <= a) result=2;
```

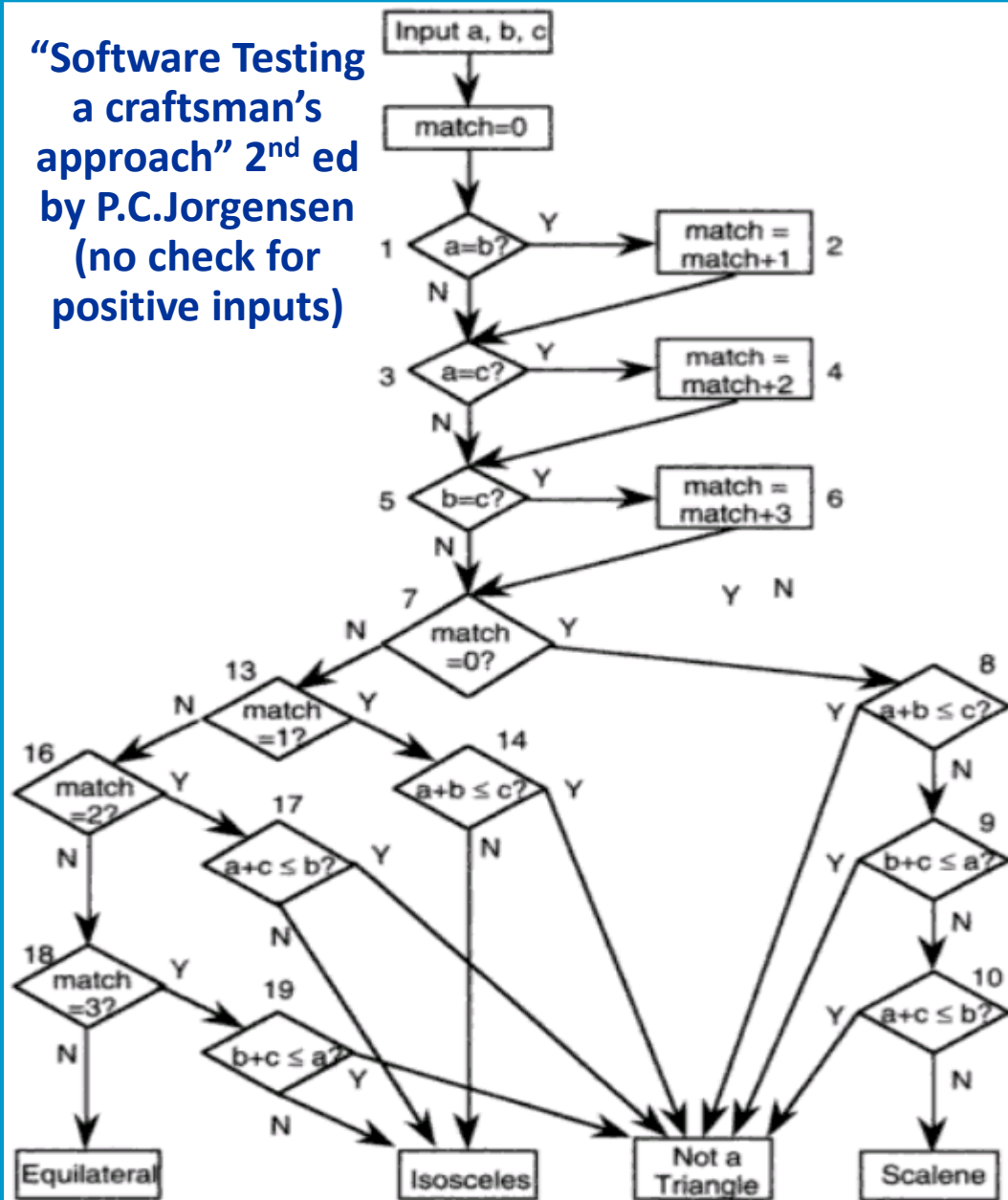
```
                    else result=1;
```

```
                } else result = 0;
```

```
            } } }
```

```
    printf("result=%d\n",result); }
```

“Software Testing
a craftsman’s
approach” 2nd ed
by P.C.Jorgensen
(no check for
positive inputs)



Concolic Testing the Triangle Program

Test case	Input (a,b,c)	Executed symbolic path condition (PC) ϕ	Next PC ψ	Solution for the next PC ψ from SMT solver
1	1,1,1	$a=b \wedge a=c \wedge b=c$	$a=b \wedge a=c \wedge b \neq c$	Unsat
			$a=b \wedge a \neq c$	1,1,2
2	1,1,2	$a=b \wedge a \neq c \wedge b \neq c \wedge a+b \leq c$	$a=b \wedge a \neq c \wedge b \neq c \wedge a+b > c$	2,2,3
3	2,2,3	$a=b \wedge a \neq c \wedge b \neq c \wedge a+b > c$	$a=b \wedge a \neq c \wedge b=c$	Unsat
			$a \neq b$	2,1,2
4	2,1,2	$a \neq b \wedge a=c \wedge b \neq c \wedge a+b > c$	$a \neq b \wedge a=c \wedge b \neq c \wedge a+b \leq c$	2,5,2

