

Primes: {4, 5, 3, 6, 8}, {1, 2, 3, 4, 5}, {1, 2, 3, 6, 7}, {1, 2, 3, 6, 8}, {3, 4, 3}, {4, 3, 4}, {3, 4, 5, 3},
{4, 5, 3, 4}, {5, 3, 4, 5}, {4, 3, 6, 7}, {4, 3, 6, 8}, {4, 5, 3, 6, 7}

DU: {1, 2, 3, 4, 5}, {1, 2, 3, 6, 8}, {1, 2, 3, 6, 7}, {5, 3, 6, 8}, {1, 2, 3, 4, 3}, {2, 3, 4}, {2, 3, 6},
{4, 5}, {1, 2}

@Test1: OrderHistory = null

$\rightarrow \text{circular} = [1 \rightarrow 2 \rightarrow 3 \rightarrow 6 \rightarrow 7] \Rightarrow \text{Primes, DU: } \{2, 3, 6\}$

@Test2: order History = {Order(customer: 0, quantity: 10), customer = 1
, Order(customer: 1, quantity: 20)}

$\text{circular} = [2, 2, 3, 4, 3, 4, 5, 3, 6, 8] \Rightarrow \text{Primes: } \{4, 5, 3, 6, 8\}, \{3, 4, 3\}, \{4, 3, 4\}, \{3, 4, 5, 3\}$
DU: {5, 3, 6, 8}, {1, 2, 3, 4, 3}, {2, 3, 4}, {4, 5}, {1, 2}

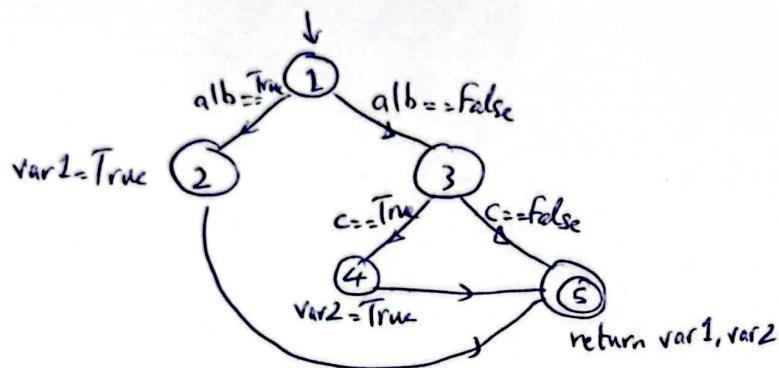
@Test3: Order History = {Order(customer: 1, quantity: 10),
, Order(customer: 1, quantity: 20),
, Order(customer: 0, quantity: 30)}, customer = 1

$\text{circular} = [1 \rightarrow 2 \rightarrow 3 \rightarrow 4, 5 \rightarrow 3 \rightarrow 4 \rightarrow 5 \rightarrow 3 \rightarrow 4 \rightarrow 3 \rightarrow 6 \rightarrow 6]$

orderHistory: [1, 2, 3, 4, 5, 3, 6, 8] \Rightarrow DU $\{1, 2, 3, 4, 5, 6, 8\}$
, customer = 1 \Rightarrow Primes: {4, 5, 3, 6, 8}, {1, 2, 3, 4, 5}, {2, 3, 4}, {4, 5}, {1, 2}

runlike $\sum / \text{Count} < i$ of `orderHistory`, \sum is the sum of `customer[i].orderCount`.
 if $a > b$ then $i = j$, otherwise $i = k$.
 if $(\text{orderHistory.size()} == 0)$
 return 0;
 if $(\text{Count} == 0)$
 return 0;

شکل ۱۲
 (الف)



Branch Coverages:

$a = \text{False}, b = \text{True}, c = \text{False} \Rightarrow 1 \rightarrow 2 \rightarrow 5$

$a = \text{False}, b = \text{False}, c = \text{True} \Rightarrow 1 \rightarrow 3 \rightarrow 4 \rightarrow 5$

$a = \text{False}, b = \text{False}, c = \text{False} \Rightarrow 1 \rightarrow 3 \rightarrow 5$

Statement Coverages:

$a = \text{False}, b = \text{True}, c = \text{False} \Rightarrow 1 \rightarrow 2 \rightarrow 5$

$a = \text{False}, b = \text{False}, c = \text{True} \Rightarrow 1 \rightarrow 3 \rightarrow 4 \rightarrow 5$

این روش

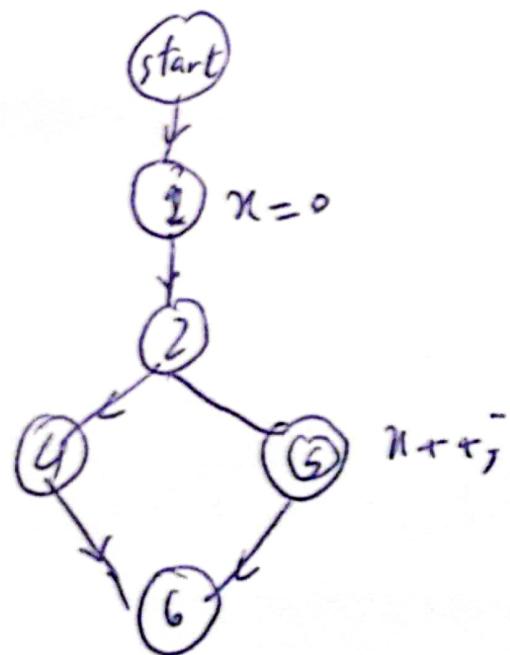
`var1 = (a | b);`

`var2 = (! (a | b)) & c;`

`return var1, var2`

(=)

CFG یعنی دایرکٹ اسٹرکچر لایبلز، branching کا code اسی سطح پر این است کہ جو کسی دو گزینے میں (سریع یا سوچنے) میں خوب این سطح لامبے اسی میں است.



سول سیم،
باراً سلسله خط خاتمه دایتی کیم

{ ۱, ۲, ۳, ۴, ۵ } DU سے

بیسیانی سے ۱, ۲, ۳, ۴, ۵ پوچش راهی سد.