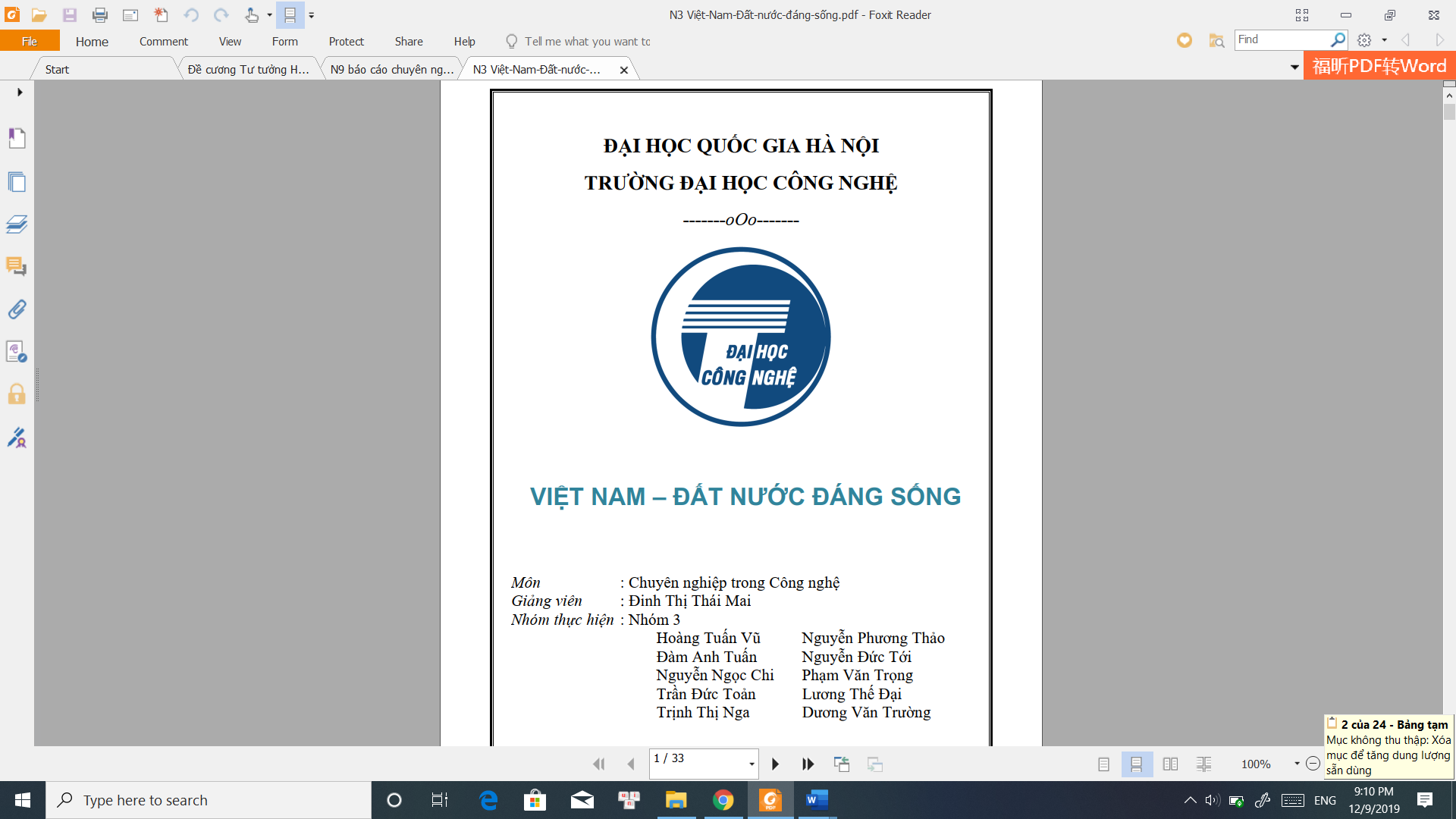
**ĐẠI HỌC QUỐC GIA HÀ NỘI  
TRƯỜNG ĐẠI HỌC CÔNG NGHỆ**

**BÀI TẬP KIỂM THỬ HỘP ĐEN**

**MÔN: KIỂM THỬ VÀ ĐẢM BẢO CHẤT LƯỢNG PHẦN MỀM**

**Giảng viên**: ThS Nguyễn Thu Trang

**Sinh viên:** Đào Minh Hoàn 18020535

**Mục lục**

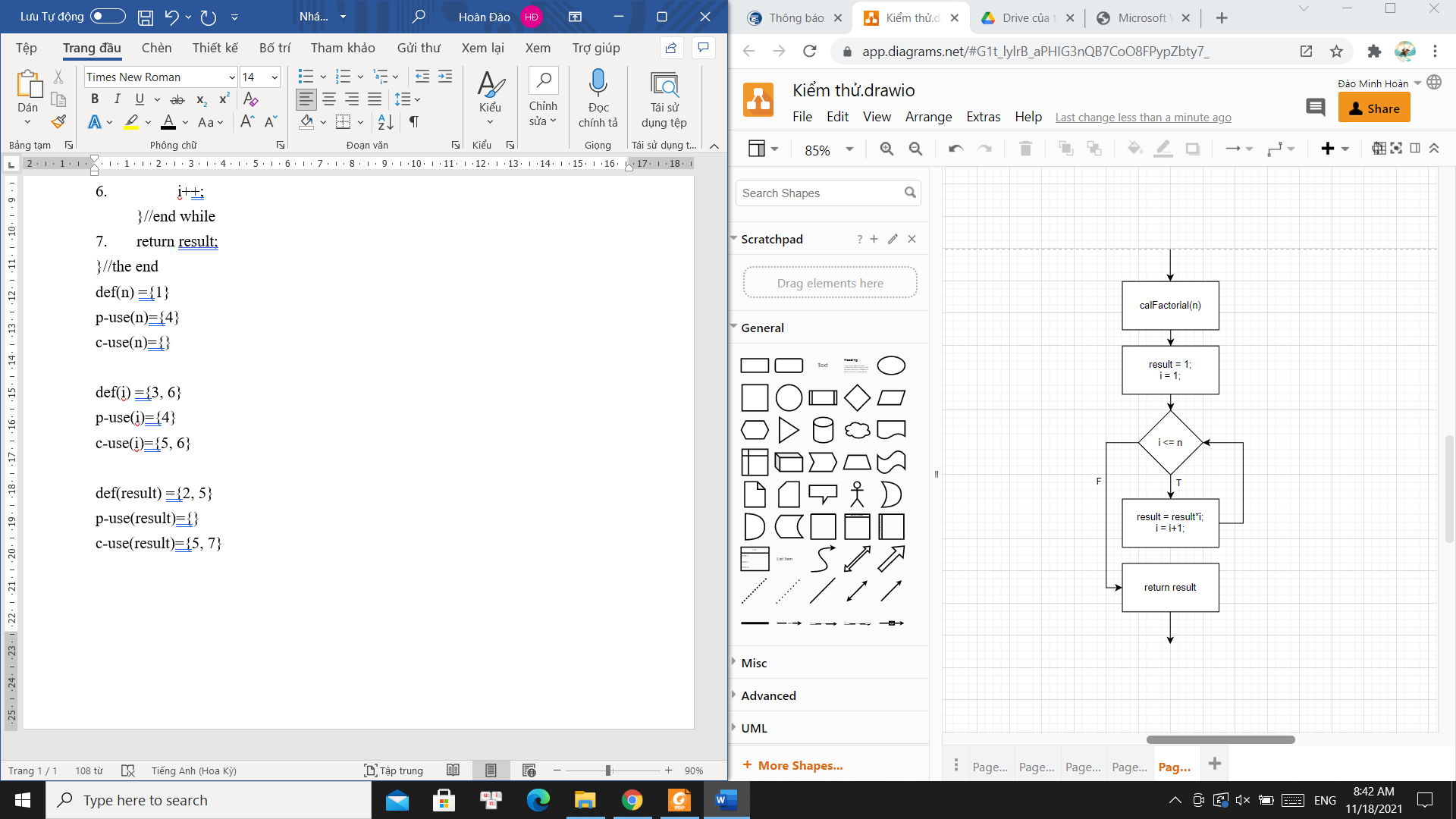
[Câu 7 1](#_Toc88229824)

[Câu 10 2](#_Toc88229825)

[Bài tập 1 4](#_Toc88229826)

[Bài tập 2 6](#_Toc88229827)

[Bài tập 3. Kiểm thử chương trình của bạn với độ phủ all-c-uses/some-p-uses 9](#_Toc88229828)

Câu 7:

1.int calFactorial (int n) {

2. int result = 1;

3. int i = 1;

4. while (i <= n) {

5. result = result \*i;

6. i++;

}//end while

7. return result;

}//the end

-Liệt kê các câu lệnh tương ứng với các khái niệm:

def(n) ={1}

p-use(n)={4}

c-use(n)={}

def(i) ={3, 6}

p-use(i)={4}

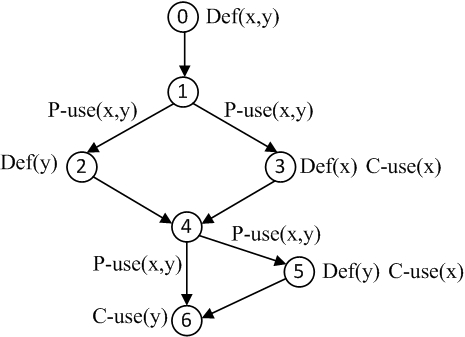
c-use(i)={5, 6}

def(result) ={2, 5}

p-use(result)={}

c-use(result)={5, 7}

## Câu 10



def(x) = {0, 3}, p-use(x) = {1, 4}, c-use(x) ={3, 5}

def(y) = {0, 2, 5}, p-use(x) = {1, 4}, c-use(y) ={6}

-Tất cả các Def-clear path và Du-pair của các biến x và y

|  |  |  |  |
| --- | --- | --- | --- |
| Variable | Du-pair | Def-clear path | Complete path |
| x | (0,1) | 0,1 | 0,1,3,4,6 |
|  | (0,4) | 0,1,2,4 | 0,1,2,4,6 |
|  | (0,3) | 0,1,3 | 0,1,3,4,6 |
|  | (0,5) | 0, 1, 2, 4, 5 | 0, 1, 2, 4, 5, 6 |
|  | (3, 4) | 3, 4 | 0, 1, 3, 4, 6 |
|  | (3, 5) | 3, 4, 5 | 0, 1, 3, 4, 5, 6 |
| y | (0, 1) | 0, 1 | 0,1,3,4,6 |
|  | (0, 4) | 0, 1, 3, 4 | 0, 1, 3, 4, 6 |
|  | (0, 6) | 0, 1, 3, 4, 6 | 0, 1, 3, 4, 6 |
|  | (2, 4) | 2, 4 | 0, 1, 2, 4, 6 |
|  | (2, 6) | 2, 4, 6 | 0, 1, 2, 4, 6 |
|  | (5, 6) | 5, 6 | 0, 1, 3, 4, 5, 6 |

-All-p-uses/Some-c-uses

|  |  |  |  |
| --- | --- | --- | --- |
| Variable | Du-pair | Def-clear path | Complete path |
| x | (0,1) | 0,1 | 0,1,3,4,6 |
|  | (0,4) | 0,1,2,4 | 0,1,2,4,6 |
|  | (3, 4) | 3, 4 | 0, 1, 3, 4, 6 |
| y | (0, 1) | 0, 1 | 0,1,3,4,6 |
|  | (0, 4) | 0, 1, 3, 4 | 0, 1, 3, 4, 6 |
|  | (2, 4) | 2, 4 | 0, 1, 2, 4, 6 |

-All-c-uses/Some-p-uses

|  |  |  |  |
| --- | --- | --- | --- |
| Variable | Du-pair | Def-clear path | Complete path |
| x | (0,3) | 0,1,3 | 0,1,3,4,6 |
|  | (0,5) | 0, 1, 2, 4, 5 | 0, 1, 2, 4, 5, 6 |
|  | (3, 5) | 3, 4, 5 | 0, 1, 3, 4, 5, 6 |
| y | (0, 6) | 0, 1, 3, 4, 6 | 0, 1, 3, 4, 6 |
|  | (2, 6) | 2, 4, 6 | 0, 1, 2, 4, 6 |
|  | (5, 6) | 5, 6 | 0, 1, 3, 4, 5, 6 |

-Biểu thức của các *p*-*use*(*x; y*) tại cạnh (1,3) và (4,5) lần lượt là *x* + *y* = 4 và x2+y2 > 17 . Đường đi (0 - 1 - 3 - 4 - 5 - 6) có thực thi được không? Giải thích

Đường đi này có thực thi được. Vì nếu ta cho x = -3, y = 7 thì lúc này x, y đều thỏa mãn x+y=4 và x2+y2 > 17.

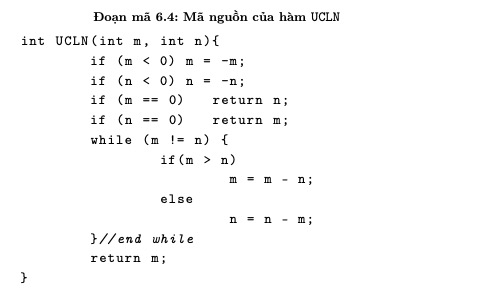
- Tại sao tại đỉnh 3 biến x được định nghĩa và sử dụng nhưng không tồn tại mối quan hệ def-use?

Theo định nghĩa:

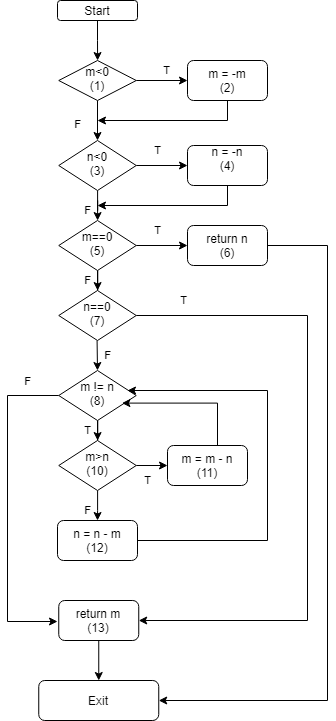
**Du-pair (Def-use pair):** (d,u) là cặp def-use của biến v nếu:  
§ d là đỉnh tương ứng với câu lệnh def của v  
§ u là đỉnh tương ứng với câu lệnh use của v  
§ Tồn tại ít nhất một def-clear path của biến v từ d tới u

Mà ở đây không tồn tại một def-clear path của biến x từ đỉnh 3 lại trở về đỉnh 3. Vì vậy không tồn tại mối quan hệ def-use

## Bài tập 1



1. Xây dựng CFG cho hàm UCLN với đồ thị C2



1. Sinh đường đi và các ca kiểm thử với độ đo C2

-Đường đi độ đo C2 (số được đánh trong đồ thị):

* Star-1(T)-2-3(T)-4-5(F)-7(F)-8(F)-13-Exit
* Star-1(F)-3(F)-5(T)-6-Exit
* Start-1(F)-3(F)-5(F)-7(T)-13-Exit
* Start-1(F)-3(F)-5(F)-7(F)-8(T)-10(T)-11-8(F)-13-Exit
* Start-1(F)-3(F)-5(F)-7(F)-8(T)-10(F)-12-8(F)-13-Exit

-Sinh ca kiểm thử cho độ đo C2:

UCLN(-1, -1), UCLN(0, 7), UCLN(7, 0), UCLN(8, 4), UCLN(4, 8)

1. Sinh đường đi và các ca kiểm thử với độ đo all-def coverage

1.int UCLN (int m, int n) {

2. if(m<0)

3. m = -m;

4. if(n<0)

5. n = -n;

6. if(m==0)

7. return n;

8. if(n==0)

9. return m;

10. while (m != n){

11. if(m>n)

12. m = m – n;

13. else

14. n = n – m;

}

15. return m;

}

-Biến m: def(m)={1, 3, 12}, p-use(m) ={2, 6, 10, 11}, c-use(m)={3, 9, 12, 14, 15}

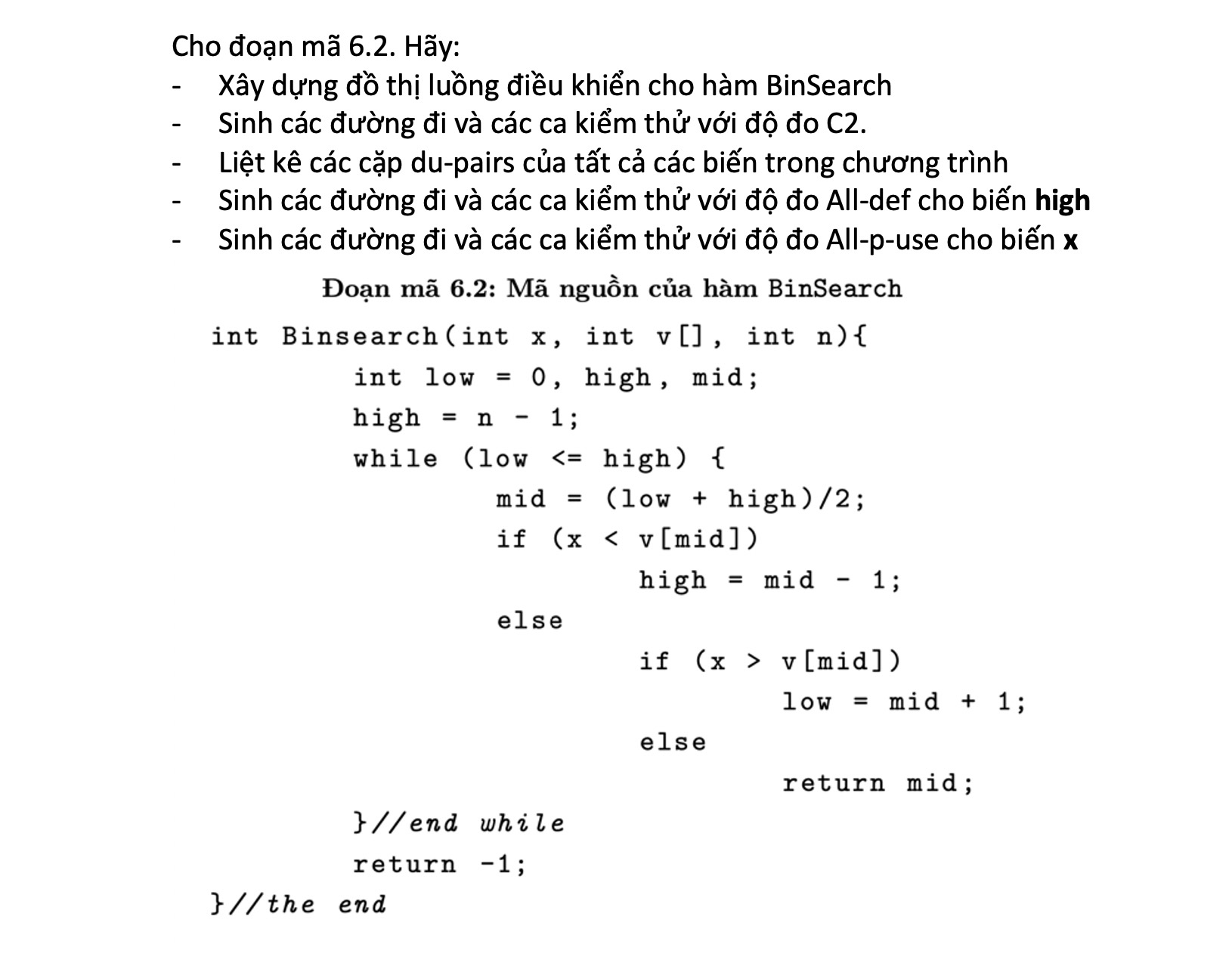
-Biến n: def(n)={1, 5, 14}, p-use(n) ={4, 8, 10, 11}, c-use(n)={5, 7, 12, 14}

|  |  |  |  |
| --- | --- | --- | --- |
| Variable | Du-pair | Def-clear path | Complete path |
| m | (1,9) | 1, 2(F), 4(F), 6(F), 8(T), 9 | 1, 2(F), 4(F), 6(F), 8(T), 9 |
|  | (3,9) | 3, 4(F), 6(F), 8(T), 9 | 1, 2(T), 3, 4(F), 6(F), 8(T), 9 |
|  | (12,12) | 12,10(T), 11(T),12 | 1, 2(F), 4(F), 6(F), 8(F), 10(T), 11(T), 12,10(T), 11(T),12, 15 |
| n | (1, 7) | 1, 2(F), 4(F), 6(T), 7 | 1, 2(F), 4(F), 6(T), 7 |
|  | (5, 7) | 5, 6(T), 7 | 1, 2(F), 4(T), 5, 6(T), 7 |
|  | (14, 14) | 14, 10(T), 11(F), 14 | 1, 2(F), 4(F), 6(F), 8(F), 10(T), 11(F), 14,10(T), 11(F),14, 15 |

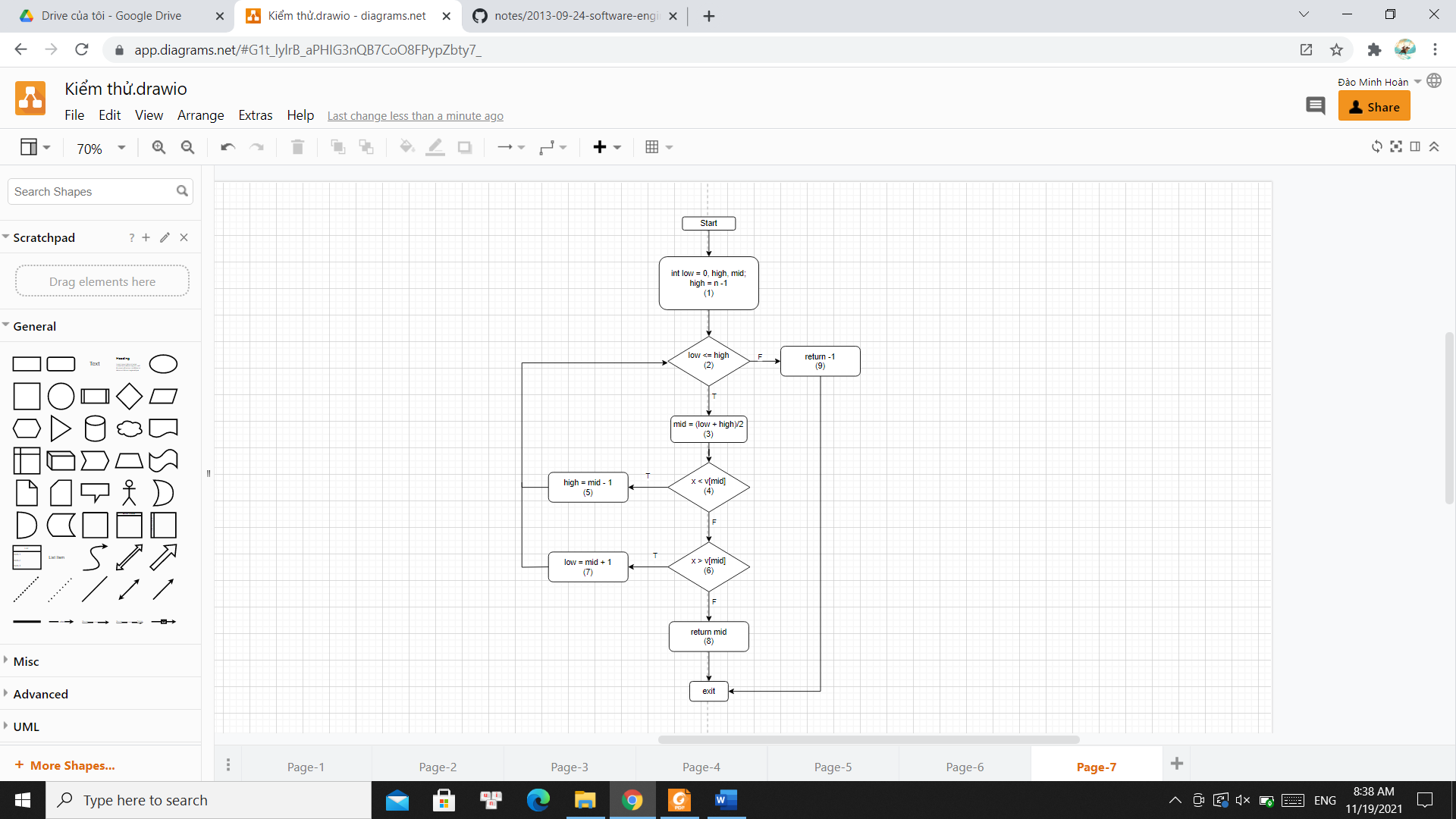
Sinh ca kiểm thử với độ đo all-def coverage

UCLN(7, 0), UCLN(-7, 0), UCLN(12, 4), UCLN(0, 7), UCLN(0, -7), UCLN(4, 12)

## Bài tập 2



1.Đồ thị luồng điều khiển



2.Sinh đường đi và các ca kiểm thử với độ đo C2

-Đường đi độ đo C2 (số được đánh trong đồ thị):

* Star-1-2(T)-3-4(T)-5-2(F)-9-Exit
* Start-1-2(T)-3-4(F)-6(F)-8-Exit
* Start-1-2(T)-3-4(F)-6(T)-7-2(F)-9-Exit

-Sinh ca kiểm thử cho độ đo C2:

Binsearch(0,[1],1), Binsearch(1, [1], 1), Binsearch(2, [1], 1)

3.Liệt kê các cặp du-pairs

1.int Binsearch (int x, int v[], int n){  
2. int low = 0, high , mid;  
3. high = n - 1;  
4. while ( low <= high ) {  
5. mid = (low + high )/2;  
6. if (x < v[ mid ])  
7. high = mid - 1;  
 else  
8. if (x > v[ mid ])  
9. low = mid + 1;  
 else  
10. return mid;  
 }*// end while*11. return -1;  
}*// the end*

-Biến x: def(x)={1}, p-use(x)={6, 8}, c-use(x)={}

-Biến v: def(v)={1}, p-use(v)={6, 8}, c-use(v)={}

-Biến n: def(n)={1}, p-use(n)={}, c-use(n)={3}

-Biến mid: def(mid)={5}, p-use(mid)={6, 8}, c-use(mid)={7, 9, 10}

-Biến low: def(low)={2, 9}, p-use(low)={4}, c-use(low)={5}

-Biến high: def(high)={3, 7}, p-use(high)={4}, c-use(high)={5}

|  |  |  |  |
| --- | --- | --- | --- |
| Variable | Du-pair | Def-clear path | Complete path |
| x | (1, 6) | 1, 2, 3, 4(T), 5, 6 | 1, 2, 3, 4(T), 5, 6(F), 8(F), 10 |
|  | (1, 8) | 1, 2, 3, 4(T), 5, 6(F), 8 | 1, 2, 3, 4(T), 5, 6(F), 8(F), 10 |
| v | (1,6) | 1, 2, 3, 4(T), 5, 6 | 1, 2, 3, 4(T), 5, 6(F), 8(F), 10 |
|  | (1,8) | 1, 2, 3, 4(T), 5, 6(F), 8 | 1, 2, 3, 4(T), 5, 6(F), 8(F), 10 |
| n | (1,3) | 1, 2, 3 | 1, 2, 3, 4(T), 5, 6(F), 8(F), 10 |
| mid | (5, 6) | 5, 6 | 1, 2, 3, 4(T), 5, 6(F), 8(F), 10 |
|  | (5, 8) | 5, 6(F), 8 | 1, 2, 3, 4(T), 5, 6(F), 8(F), 10 |
|  | (5, 7) | 5, 6(T), 7 | 1, 2, 3, 4(T), 5, 6(T), 7, 4(F), 11 |
|  | (5, 9) | 5, 6(F), 8(T), 9 | 1, 2, 3, 4(T), 5, 6(F), 8(T), 9, 4(F), 11 |
|  | (5, 10) | 5, 6(F), 8(F), 10 | 1, 2, 3, 4(T), 5, 6(F), 8(F), 10 |
| low | (2, 4) | 2, 3, 4 | 1, 2, 3, 4(T), 5, 6(F), 8(F), 10 |
|  | (2, 5) | 2, 3, 4(T), 5 | 1, 2, 3, 4(T), 5, 6(F), 8(F), 10 |
|  | (9, 4) | 9, 4 | 1, 2, 3, 4(T), 5, 6(F), 8(T), 9, 4(F), 11 |
|  | (9, 5) | 9, 4(T), 5 | 1, 2, 3, 4(T), 5, 6(F), 8(T), 9, 4(T), 5, 6(F), 8(F), 10 |
| high | (3, 4) | 3, 4 | 1, 2, 3, 4(T), 5, 6(F), 8(F), 10 |
|  | (3, 5) | 3, 4(T), 5 | 1, 2, 3, 4(T), 5, 6(F), 8(F), 10 |
|  | (7, 4) | 7, 4 | 1, 2, 3, 4(T), 5, 6(T), 7, 4(F), 11 |
|  | (7, 5) | 7, 4(T), 5 | 1, 2, 3, 4(T), 5, 6(T), 7, 4(T), 5, 6(T), 7, 4(F), 11 |

4.Độ đo All-def cho biến high

-Biến high: def(high)={3, 7}, p-use(high)={4}, c-use(high)={5}

|  |  |  |  |
| --- | --- | --- | --- |
| Variable | Du-pair | Def-clear path | Complete path |
| high | (3, 4) | 3, 4 | 1, 2, 3, 4(T), 5, 6(F), 8(F), 10 |
|  | (7, 5) | 7, 4(T), 5 | 1, 2, 3, 4(T), 5, 6(T), 7, 4(T), 5, 6(T), 7, 4(F), 11 |

Sinh ca kiểm thử với độ đo All-def cho biến high

Binsearch(1, [1], 1), Binsearch(0, [2, 1, 3], 3)

5.Độ đo All-p-use cho biến x

-Biến x: def(x)={1}, p-use(x)={6, 8}, c-use(x)={}

|  |  |  |  |
| --- | --- | --- | --- |
| Variable | Du-pair | Def-clear path | Complete path |
| x | (1, 6(F)) | 1, 2, 3, 4(T), 5, 6(F) | 1, 2, 3, 4(T), 5, 6(F), 8(F), 10 |
|  | (1, 6(T)) | 1, 2, 3, 4(T), 5, 6(T) | 1, 2, 3, 4(T), 5, 6(T), 7, 4(F), 11 |
|  | (1, 8(F)) | 1, 2, 3, 4(T), 5, 6(F), 8(F) | 1, 2, 3, 4(T), 5, 6(F), 8(F), 10 => Giống với (1, 6(F)) |
|  | (1, 8(T)) | 1, 2, 3, 4(T), 5, 6(F), 8(T) | 1, 2, 3, 4(T), 5, 6(F), 8(T), 9, 4(F), 11 |

Sinh ca kiểm thử với độ đo All-p-uses cho biến x

Binsearch(1, [1], 1), Binsearch(1, [2], 1), Binsearch(2, [1], 1)

## Bài tập 3. Kiểm thử chương trình của bạn với độ phủ all-c-uses/some-p-uses

1.public String CheckSale(int k, int b, int n) {

2. int sum = 35\*k + 20\*b + 15\*n;

3. if(!(k>=0 && k<=70) || !( b>=0 && b<=80) || !( n>=0 && n<=90) )

4. return "Invalid";

5. if(k < 3 || b < 3 || n < 3 || sum <= 250)

6. return "No";

7. return "Yes";

}

-Biến k: def(k)={1}, p-use(k)={3, 5}, c-use(k)={2}

-Biến b: def(b)={1}, p-use(x)={3, 5}, c-use(x)={2}

-Biến n: def(n)={1}, p-use(x)={3, 5}, c-use(x)={2}

-Biến sum: def(sum)={2}, p-use(sum)={5}, c-use(sum)={}

Độ phủ all-c-uses/some-p-uses

|  |  |  |  |
| --- | --- | --- | --- |
| Variable | Du-pair | Def-clear path | Complete path |
| k | (1, 2) | 1, 2 | 1, 2, 3(F), 4 |
| b | (1, 2) | 1, 2 | 1, 2, 3(F), 4 |
| n | (1, 2) | 1, 2 | 1, 2, 3(F), 4 |
| sum | (2, 5) | 2, 3(F), 5 | 1, 2, 3(F), 5(T), 6 |

Sinh các ca kiểm thử với độ đo all-c-uses/some-p-uses

CheckSale(-1, 45, 30), CheckSale(0, 81, 30), CheckSale(0, 2, -3), CheckSale(1, 1, 1)