Câu hỏi 2.15: Cho Form gồm 1 ratio button Nữ, 1 ratio button Độc thân và 1 Danh sách chọn Chuyên ngành gồm: CNTT, QTKD, VT, Kế toán.

a. Nếu kiểm thử tất cả các trường hợp xảy ra thì cần bao nhiêu test cases,

b. Mỗi test case chứa bao nhiêu cặp giá trị?

c. Liệt kê các cặp giá trị có thể xảy ra?

d. Thiết kế bộ pairwise test suite (kiểm thử theo cặp)

Bài làm:

1. Nếu kiểm thử tất cả các trường hợp xảy ra thì cần 2 \*2 \* 4 = 16 test cases
2. Số lượng cặp giá trị trong mỗi test case 3\*2/2 = 3 cặp giá trị
3. Các cặp giá trị có thể xảy ra:

(button Nữ/ button Độc thân) = {Y/N, N/Y, N/N, Y/Y}

(button Nữ/ Chuyên ngành gồm) ={Y/CNTT, Y/QTKD, ....}

(button Độc thân / Chuyên ngành gồm) = {Y/CNTT, Y/QTKD, ....}



|  |  |  |
| --- | --- | --- |
| button Nữ | button Độc thân | Chuyên ngành |
| YES | YES | CNTT |
| YES | NO | QTKD |
| NO | YES | VT |
| NO | NO | Kế toán |
| NO | NO | CNTT |
| NO | YES | QTKD |
| YES | NO | VT |
| YES | YES | Kế toán |

Câu hỏi 2.16: Một phần mềm điều khiển chơi game đơn giản giữa 2 người chơi. Mỗi người điều khiển một nút bấm để đưa bóng vào lỗ. Mỗi lần bóng vào lỗ, người đó được cộng 1 điểm. Ai được 5 điểm trước sẽ thắng. Nếu bóng không vào lỗ, người còn lại được quyền điều khiẻn bóng.

a. Lập sơ đồ chuyển trạng thái

b. Xác định các đường chạy để phủ hết các cạnh

c. Thiết kế test case tương ứng

Câu hỏi 2.17: Thông tin về block1 bao gồm size(small, large), color(red, green, blue), shape (circle, triangle, square).

a. Nếu kiểm thử tất cả các trường hợp xảy ra thì cần bao nhiêu ca kiểm thử?

b. Số cặp tối đa mà một ca kiểm thử có thể chứa

c. Xác định các cặp giá trị có thể xảy ra

d. Thiết kế bộ kiểm thử theo cặp (pairwise test suite)

Bài làm:

1. Nếu kiểm thử tất cả các trường hợp xảy ra thì cần 2\*3\*3 = 18 ca kiểm thử
2. Số cặp tối đa mà một ca kiểm thử có thể chứa là 3\*2/2 = 3 cặp
3. Xác định các cặp giá trị có thể xảy ra

* (size / color) ={(small/red), (small/blue), (large/red), ....}
* (size/ shape) ={(small/circle), (small/triagle), (large/square), ....}
* (color/ shape) ={(red/circle), (red/triagle), (green/square), ....}



|  |  |  |
| --- | --- | --- |
| SIZE | COLOR | SHAPE |
| Small | Red | Circle |
| Large | Red | Triagle |
| Small | Red | square |
| Large | Green | Circle |
| Small | Green | Triagle |
| Large | Green | square |
| Small | blue | Circle |
| Large | blue | Triagle |
| Small | blue | square |
|  |  |  |

Câu hỏi 2.18: Cần phát triển module kiểm tra điều kiện dự thi của sinh viên gồm: Nếu sinh viên đi học >=80% số buổi, điểm giữa kì >0, điểm bài tập lớn >0 sẽ được thi. Nếu sinh viên đủ điều kiện dự thi và có điểm bài tập lớn = 10 hoặc điểm giữa kỳ = 10 sẽ được miễn thi.

a. Dùng kỹ thuật bảng quyết định để xác định test cases

b. Dùng kỹ thuật đồ thị nguyên nhân kết quả để xác định test cases



|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Rule 1 | Rule 2 | Rule 3 | Rule 4 | Rule 5 | Rule 6 | Rule 7 |  |
| **Condittions** |  |  |  |  |  |  |  |  |
| Đi học>80% | y | y | y | y | y |  |  |  |
| Điểm giữa kì=0 |  | y |  |  |  |  |  |  |
| 0<Điểm giữa kì<10 | y |  | y |  | y |  |  |  |
| Điểm giữa kì = 10 |  |  |  | y |  |  |  |  |
| Điểm BTL=0 |  |  | y |  |  |  |  |  |
| 0<Điểm BTL <10 | y | y |  | y |  |  |  |  |
| Điểm BTL= 10 |  |  |  |  | y |  |  |  |
| **Actions** |  |  |  |  |  |  |  |  |
| Được thi | y |  |  |  |  |  |  |  |
| Không được thi |  | y | y |  |  |  |  |  |
| Miễn thi |  |  |  | y | y |  |  |  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | ĐI học >= 80% | Điểm giữa kỳ >0 | Điểm btl >0 | Điểm giữa kỳ =10 | Điểm BTL =10 | Được thi | Không được thi | Miễn thi |
| Rule 1 | T | T | T | T | T | T |  | T |
| Rule 2 | T | T | T | T | F | T |  | T |
| Rule 3 | T | T | T | F | T | T |  | T |
| Rule 4 | T | T | T | F | F | T |  |  |
| Rule 5 | T | T | F | - | - |  | T |  |
| Rule 6 | T | F | T | - | - |  | T |  |
| Rule 7 | T | F | F | - | - |  | T |  |
| Rule 8 | F | - | - | - | - |  | T |  |

|  |  |  |
| --- | --- | --- |
| TC | Đầu vào | Kết quả mong đợi |
| Rule 1 | Đi học > 80% điểm giữa kỳ bằng 10 điểm BTL =10 | Được miễn thi |
| Rule 2 | Đi học > 80% điểm giữa kỳ >0 điểm BTL =10 | Được miễn thi |
| Rule 3 | Đi học > 80% điểm giữa kỳ =10 điểm BTL >0 | Được miễn thi |
| Rule 4 | Đi học > 80% điểm giữa kỳ >0, điểm BTL >0 | Được thi |
| Rule 5 | Đi học > 80% điểm giữa kỳ >0, điểm BTL <=0 | Khong duoc thi |
| Rule 6 | Đi học > 80% điểm giữa kỳ <=0, điểm BTL >0 | Khong duoc thi |
| Rule 7 | Đi học > 80% điểm giữa kỳ <=0, điểm BTL <=0 | Khong duoc thi |
| Rule 8 | Đi học <= 80% | Khong duoc thi |
|  |  |  |

Câu hỏi 2.19: Cho hệ thống S nhận n tham số đầu vào, mỗi tham số có m giá trị. Trả lời câu hỏi:

a. Số cặp tối đa mà một ca kiểm thử chứa tối đa bao nhiêu cặp

b. Trong trường hợp lý tưởng, ta cần bao nhiêu ca kiểm thử để bao phủ tất cả các cặp của hệ thống?

c. Tính tổng số cặp mà bộ kiểm thử phải bao phủ?

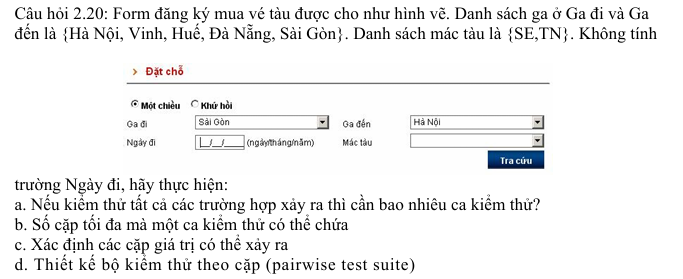
d. Cho n = 13, m = 3. Số ca kiểm thử tối thiểu cần chọn để thu được bộ kiểm thử theo cặp (pairwise test suite)?

a. Số cặp tối đa mà một ca kiểm thử chứa tối đa bao nhiêu cặp : n\*(n-1)/2 cặp

b. Trong trường hợp lý tưởng, ta cần bao nhiêu ca kiểm thử để bao phủ tất cả các cặp của hệ thống? n\*(n-1)/2 \* m (ca)

c. Tính tổng số cặp mà bộ kiểm thử phải bao phủ? n\*(n-1)/2 \* m^2

d. n\*(n-1)/2 \* m = 13\*12/2 \* 3 = 234



1. Nếu kiểm thử tất cả các trường hợp thì cần 2 \* 5 \* 4 \* 2 = 80 ca
2. Số cặp tối đa mà một ca có thể chứa = 4\*3/2 = 6 cặp

(Loại vé/Ga đi)={(Một chiều/Sài Gòn),(Một chiều/ Đà Nẵng),......}

(Loại vé/Ga đến)={(Một chiều/Sài Gòn),(Một chiều /Đà Nẵng),......}

(Loại vé/Mác tàu)={(Một chiều/SE),(Một chiều/TN),......}

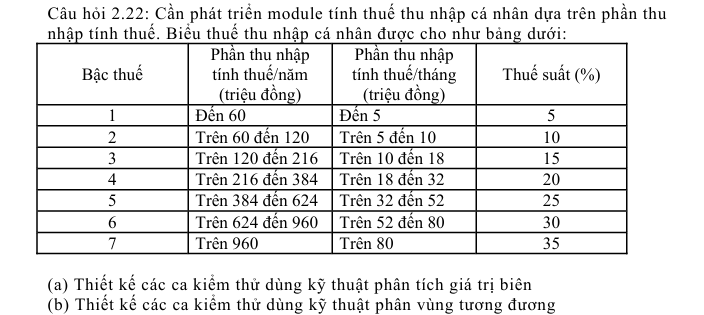
(Ga đi/Ga đến)={(Vinh/Sài Gòn),(Huế/Đà Nẵng),......}

(Ga đi/Mác tàu)={(Vinh/SE),(Sài Gòn/ TN),......}

(Ga đến/Mác tàu)={(Vinh/SE),(Sài Gòn/ TN),......}



|  |  |  |  |
| --- | --- | --- | --- |
| Loại vé | Ga đi | Ga đến | Mác tàu |
| Một chiều | Hà Nội | Vinh | SE |
| Một chiều | Hà Nội | Huế | TN |
| Khứ hồi | Hà Nội | Đà Nẵng | SE |
| Khứ hồi | Hà Nội | Sài Gòn | TN |
| Một chiều | Vinh | Hà Nội | SE |
| Một chiều | Vinh | Huế | TN |
| Khứ hồi | Vinh | Đà Nẵng | SE |
| Khứ hồi | Vinh | Sài gòn | TN |
| Một chiều | Huế | Hà Nội | SE |
| Một chiều | Huế | Vinh | TN |
| Khứ hồi | Huế | Đà Nẵng | SE |
| Khứ hồi | Huế | Sài Gòn | TN |
| Một chiều | Đà Nẵng | Hà Nội | SE |
| Một chiều | Đà Nẵng | Vinh | TN |
| Khứ hồi | Đà Nẵng | Huế | SE |
| Khứ hồi | Đà Nẵng | Sài Gòn | TN |
| Một chiều | Sài Gòn | Hà Nội | SE |
| Một chiều | Sài Gòn | Vinh | TN |
| Khứ hồi | Sài Gòn | Huế | SE |
| Khứ hồi | Sài Gòn | Đà Nẵng | TN |



1. Phân tích giá trị biên

+ 0 < thu nhập mỗi nằm <=60 -> biên 60, biên 0

+: 60 <thu nhập mỗi nằm <=120 -> biên 120, 61

+ 120<thu nhập mỗi nằm <=216 -> biên 121, 216

+ 216<thu nhập mỗi nằm <=384 -> biên 217, 384

+ 384<thu nhập mỗi nằm <=624-> biên 385, 624

+ 624<thu nhập mỗi nằm <=960-> biên 625, 960

+ thu nhập mỗi nằm >960 -> biên 961

|  |  |  |
| --- | --- | --- |
| Test case | Input | Kết quả mong muốn |
| 1 | 0 | Invalid |
| 2 | 60 | 60\*0,05 |
| 3 | 61 | 60\*0,05 + 1\*0.1 |
| 4 | 120 | 60\*0,05 + 60\*0.1 |
| 5 | 121 | 60\*0,05 + 60\*0.1 + 1 \* 0,15 |
| 6 | 216 | 60\*0,05 + 60\*0.1 + 96 \* 0,15 |
| 7 | 217 | 60\*0,05 + 60\*0.1 + 96 \* 0,15 + 1 \*0,2 |
| 8 | 384 | 60\*0,05 + 60\*0.1 + 96 \* 0,15 + 168 \*0,2 |
| 9 | 585 | 60\*0,05 + 60\*0.1 + 96 \* 0,15 + 168 \*0,2 + 1\* 0.25 |
| 10 | 624 | 60\*0,05 + 60\*0.1 + 96 \* 0,15 + 168 \*0,2 + 240\* 0.25 |
| 11 | 625 | 60\*0,05 + 60\*0.1 + 96 \* 0,15 + 168 \*0,2 + 240\* 0.25 + 1\*0.3 |
| 12 | 960 | 60\*0,05 + 60\*0.1 + 96 \* 0,15 + 168 \*0,2 + 240\* 0.25 + 336\*0.3 |
| 13 | 961 | 60\*0,05 + 60\*0.1 + 96 \* 0,15 + 168 \*0,2 + 240\* 0.25 + 336\*0.3 + 1\*0.35 |

1. Phân tích lớp tương đương

Valid

+ 0<thu nhập mỗi nằm <=60

+ : 60 <thu nhập mỗi nằm <=120

+: 120<thu nhập mỗi nằm <=216

+ 216<thu nhập mỗi nằm <=384

+ 384<thu nhập mỗi nằm <=624

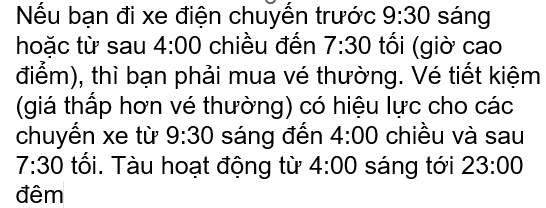
+ 624<thu nhập mỗi nằm <=960

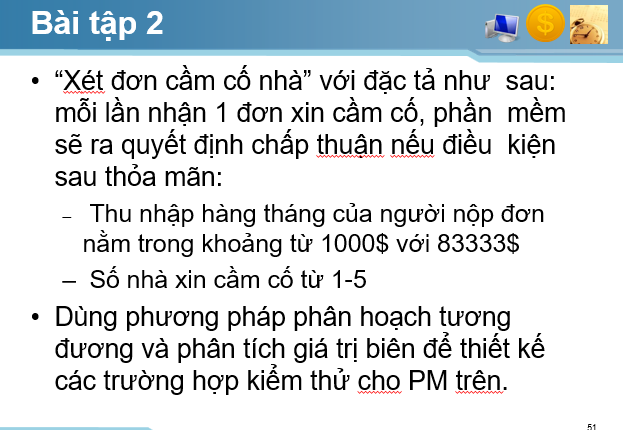
+:thu nhập mỗi nằm >960

Invalid

+ Thu nhập mỗi năm <=0

|  |  |  |
| --- | --- | --- |
| Test case | Input | Kết quả mong muốn |
| 1 | 0 | Invalid |
| 2 | 60 | 60\*0,05 |
| 3 | 61 | 60\*0,05 + 1\*0.1 |
| 5 | 121 | 60\*0,05 + 60\*0.1 + 1 \* 0,15 |
| 7 | 217 | 60\*0,05 + 60\*0.1 + 96 \* 0,15 + 1 \*0,2 |
| 9 | 585 | 60\*0,05 + 60\*0.1 + 96 \* 0,15 + 168 \*0,2 + 1\* 0.25 |
| 10 | 624 | 60\*0,05 + 60\*0.1 + 96 \* 0,15 + 168 \*0,2 + 240\* 0.25 |
| 11 | 625 | 60\*0,05 + 60\*0.1 + 96 \* 0,15 + 168 \*0,2 + 240\* 0.25 + 1\*0.3 |
| 13 | 961 | 60\*0,05 + 60\*0.1 + 96 \* 0,15 + 168 \*0,2 + 240\* 0.25 + 336\*0.3 + 1\*0.35 |





1. Nguoi 1 da ban bong

Nguoi 1 ban bong

,so diem 2 ng la 0

So diem <5, nguoi 1 ban bong

Nguoi 1 ban bong

Bong truot, diem khong doi

Bong vao lo, diem +1

2.Nguoi1 Ban da vao lo

Nguoi 1 Ban da khong vao lo

So diem nguoi 2 bang 5

Nguoi 2 ban bong

Bong truot, diem khong doi

Bong vao lo, diem +1

Nguoi 2 Ban da vao lo

Nguoi 2 Ban da khong vao lo

So diem nguoi 1bang 5

Nguoi 2 da ban bong

So diem nguoi i+1,diem nguoi I <5, nguoi I ban bong

2.Bong da vao lo

Nguoi I ban bong, I=1,so diem nguoi I = 0

1. Nguoi i da ban bong

I=2, nguoi I ban bong

Bong vao lo

So diem nguoi i+1,diem nguoi I =5,nguoi I chien than

4.Nguoi I da chien thang

Bong k vao lo

3.Bong da k vao lo

1. Xác định các đường chạy để phủ hết các cạnh

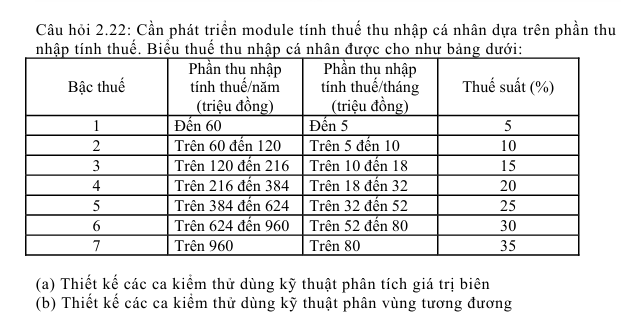
1 – 2 -4

1 – 2 – 1 -3 -1 -2 -4

1 – 3 -1 – 2- 4

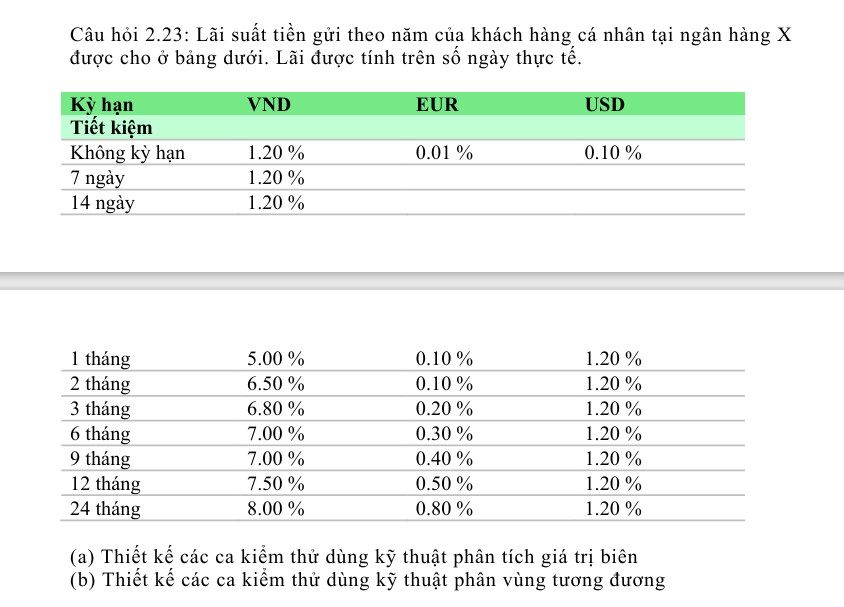
1. 1 – 2 -4

|  |  |  |  |
| --- | --- | --- | --- |
| Test case | Du lieu | Ket qua mong doi |  |
| 1-2-4 | I=1, a[i] =4 | Nguoi 1 chien thang |  |
| 1 – 2 – 1 -3 -1 -2 -4 |  |  |  |



1. Biên

Bậc thuế 1



1. Phân tích biên
2. Phân vùng tương đương

* Loại tiền gửi:
  + Valid:
    - VND
    - EUR
    - USD
* Invalid
  + Loại tiền khác
* Kỳ hạn:
  + Valid:
    - 7 ngày
    - 14 ngày
    - 1 tháng
    - ...
* Invalid
  + Những cái không trong danh sách kỳ hạn
* Số ngày gửi thực tế
  + Valid:
    - Đúng kỳ han
    - Trước kỳ hạn
    - Sau kỳ hạn
* Invalid
  + Số âm
* Số tiền gửi:
  + Valid
    - Lớn hon 0
* Invalid
  + Nhỏ hơn = 0

|  |  |  |
| --- | --- | --- |
| Test case | Dữ liệu đầu vao | KQMD |
| 1 | Loại tiền vnd |  |
|  |  |  |
|  |  |  |
|  |  |  |

Câu hỏi 3.1: Cho hàm tìm kiếm nhị phân viết bằng C. input array v đã được sắp xếp theo giá

trị tăng dần, n là kích thước mảng, ta cần tìm chỉ số mảng của phần tử x. Nếu không tìm thấy

x trong mảng, trả về giá trị -1.

int binSearch(int x, int v[], int n){

int low, high, mid;

low = 0;

high = n - 1;

while (low<=high){

mid = (low + high)/2;

if (x<v[mid])

high = mid - 1;

else if (x > v[mid])

low = mid + 1;

else

return mid;

}

return -1;

}

a. Vẽ đồ thị luồng điều khiển

b. Từ đồ thị luồng điều khiển, xác định tập các đường từ đầu vào tới đầu ra để bao phủ được

toàn bộ câu lệnh

c. Bổ xung thêm đường (nếu cần) để bao phủ hết các ngã rẽ (branch)

d. Với mỗi đường xác định ở trên, tìm biểu thức tiền tố tương ứng

e. Giải biểu thức tiền tố trên để sinh ra các đầu vào ca kiểm thử và sau đó ước lượng đầu ra

tương ứng

f. Liệu tất cả các đường trên có khả thi hay không? Nếu không chỉ ra những đường không

khả thi.

a.

1. int binSearch(int x, int v[], int n){
2. int low, high, mid;
3. low = 0;
4. high = n - 1;
5. while (low<=high){
6. mid = (low + high)/2;
7. if (x<v[mid])
8. high = mid - 1;
9. else if (x > v[mid])
10. low = mid + 1;
11. else
12. return mid;
13. }

1

1. return -1;

2

1. }

3

F

T

F

T

5

F

T

4

b. Co 4 duong

1-2-3-4-5-14-15

1-2-3-4-5-6-7-8-13-5-14-15

1-2-3-4-5-6-7-9-10-13-5-14-15

1-2-3-4-5—6-7-9-11-12-15

1. Test case

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test case | Dau vao | Ket qua mong doi | Ket qua thuc te | Result |
| 1-2-3-4-5-14-15 | X=10  V=[]  N=0 | -1 | -1 | Pass |
| 1-2-3-4-5-6-7-8-13-5-14-15 | X=1  V={1,3,5 }  N=3 | 0 | 0 | Pass |
| 1-2-3-4-5-6-7-9-10-13-5-14-15 | X=5  V={1,3,5 }  N=3 | 2 | 2 | Passs |
| 1-2-3-4-5—6-7-9-11-12-15 | X=3  V={1,3,5 }  N=3 | 1 | 1 | Pass |

Câu hỏi 3.2: Giả mã bên dưới tính tổng các phần tử >0 của mảng a

sum\_of\_all\_positive\_numbers(a, num\_of\_entries, sum)

sum =0;

init = 1;

while(init <= num\_of\_entries)

if a[init] > 0

sum = sum + a[init]

endif

init = init + 1

endwhile

end sum\_of\_all\_positive\_numbers

a. Vẽ đồ thị luồng điều khiển

b. Từ đồ thị luồng điều khiển, xác định tập các đường từ đầu vào tới đầu ra để bao phủ được

toàn bộ câu lệnh

c. Bổ xung thêm đường (nếu cần) để bao phủ hết các ngã rẽ (branch)

d. Với mỗi đường xác định ở trên, tìm biểu thức tiền tố tương ứng

e. Giải biểu thức tiền tố trên để sinh ra đầu vào các ca kiểm thử và sau đó ước lượng đầu ra

tương ứng

f. Liệu tất cả các đường trên có khả thi hay không? Nếu không chỉ ra những đường không

khả thi.

1. sum\_of\_all\_positive\_numbers(a, num\_of\_entries, sum)

1. sum =0;
2. init = 1;
3. while(init <= num\_of\_entries)
4. if a[init] > 0
5. sum = sum + a[init]
6. endif
7. init = init + 1
8. endwhile
9. end sum\_of\_all\_positive\_numbers

10

9

8

7

6

5

4

3

1

2



1-2-3-4-10

1-2-3-4-5-7-8-9-10

1-2-3-4-5-6-7-8-9-10

1-2-3-4-5-6-7-8-4-5-7-8-9-10

1. Test case

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test case | Du lieu vao | Ket qua mong doi | Kq thuc te | Result |
| 1-2-3-4-10 | A={},  num\_of\_entries=0, | 0 | 0 | Pass |
| 1-2-3-4-5-7-8-9-10 | A={-1},  num\_of\_entries=1, | 0 | 0 | Pass |
| 1-2-3-4-5-6-7-8-9-10 | A={1},  num\_of\_entries=1, | 1 | 0 | Fail |
| 1-2-3-4-5-6-7-8-4-5-6-7-8-9-10 | A={1,-1},  num\_of\_entries=1, | 1 | 0 | Fail |

Test 3 fail do init bat dau = 1, sua lai bat dau =0

Câu hỏi 3.3: Hàm bên dưới trả về chỉ số phần tử cuối cùng trong x có giá trị bằng y. Nếu

không tồn tại, trả về giá trị -1.

int findLast(int[] x, int y){

for (int i = x.length -1; i > 0; i--){

if (x[i] == y)

return i;

}

return -1;

}

a. Vẽ đồ thị luồng điều khiển

b. Từ đồ thị luồng điều khiển, xác định tập các đường từ đầu vào tới đầu ra để bao phủ được

toàn bộ câu lệnh

c. Bổ xung thêm đường (nếu cần) để bao phủ hết các ngã rẽ (branch)

d. Với mỗi đường xác định ở trên, tìm biểu thức tiền tố tương ứng

e. Giải biểu thức tiền tố trên để sinh ra đầu vào các ca kiểm thử và sau đó ước lượng đầu ra

tương ứng

f. Liệu tất cả các đường trên có khả thi hay không? Nếu không chỉ ra những đường không

khả thi.

1. int findLast(int[] x, int y){
2. for ( 2.1:int i = x.length -1; 2.2:i > 0; 2.3:i--){
3. if (x[i] == y)
4. return i;
5. }
6. return -1;
7. }

F

T

F

T



1-2.1-2.2-3-4-7

1-2.1-2.2-6-7

1. 2.1 - 2.2 – 3 - 5 - 2.3- 2.2 - 6 -7

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Testcase | Du lieu dau vao | Ket qua mong doi | Ket qua thuc te | Result |
| 1-2.1-2.2-6-7 | X={}, y=10 | -1 | -1 | Pass |
| 1-2.1-2.2-3-4-7 | X={1,2,10}, y =10 | 2 | 2 | Pass |
| 1-2.1 - 2.2 – 3 - 5 - 2.3- 2.2 - 6 -7 | X={10,20} y=10 | 0 | -1 | Fail |

Test 1-2.1 - 2.2 – 3 - 5 - 2.3- 2.2 - 6 -7 bi Fail do code khong duyet den phan tu 0 cua mang X

\

Câu hỏi 3.6: Cho đoạn code

public static void f1 (int x, int y) {

if (x < y) { f2 (y); }

else { f3 (y); };

}

public static void f2 (int a) {

if (a % 2 == 0) {

f3 (2\*a);

};

}

public static void f3 (int b) {

if (b > 0) { f4(); }

else { f5(); };

}

public static void f4()

{... f6()....}

public static void f5()

{... f6()....}

public static void f6()

{...}

a. Vẽ đồ thị luồng điều khiển

b. Từ đồ thị luồng điều khiển, xác định tập các đường từ đầu vào tới đầu ra để bao phủ được

toàn bộ câu lệnh

c. Bổ xung thêm đường (nếu cần) để bao phủ hết các ngã rẽ (branch)

d. Với mỗi đường xác định ở trên, tìm biểu thức tiền tố tương ứng

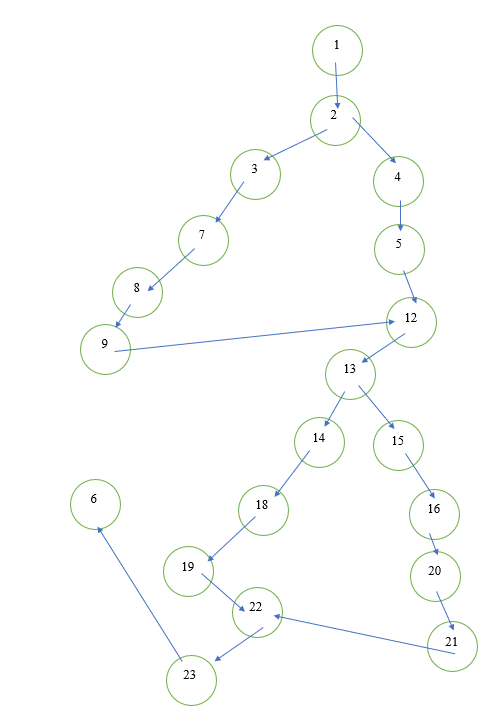
e. Giải biểu thức tiền tố trên để sinh ra đầu vào các ca kiểm thử và sau đó ước lượng đầu ra

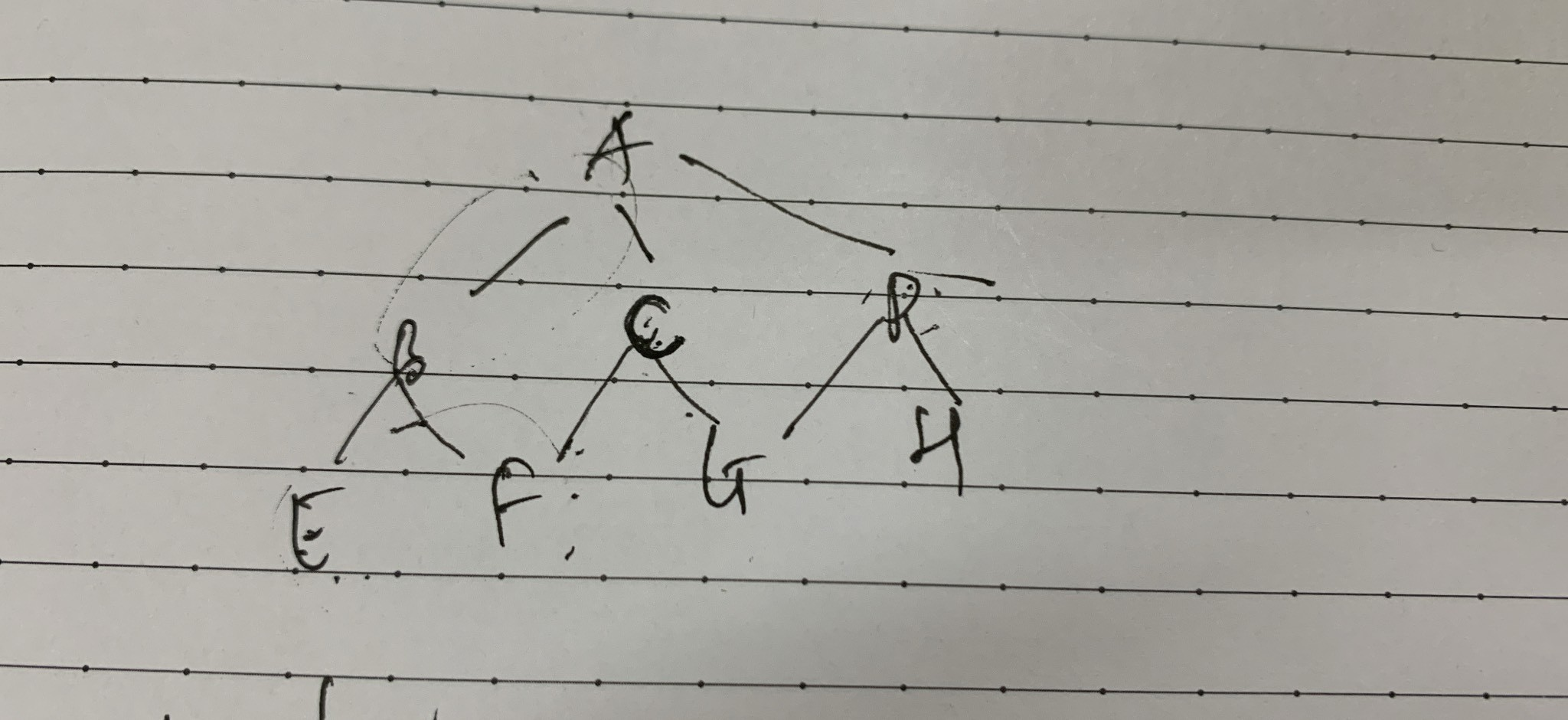
tương ứng

f. Liệu tất cả các đường trên có khả thi hay không? Nếu không chỉ ra những đường không

khả thi.

1. public static void f1 (int x, int y) {
2. if (x < y)
3. { f2 (y); }
4. else
5. { f3 (y); };
6. }
7. public static void f2 (int a) {
8. if (a % 2 == 0) {
9. f3 (2\*a);
10. };
11. }
12. public static void f3 (int b) {
13. if (b > 0)
14. { f4(); }
15. else
16. { f5(); };
17. }
18. public static void f4()
19. {... f6()....}
20. public static void f5()
21. {... f6()....}
22. public static void f6()
23. {...}





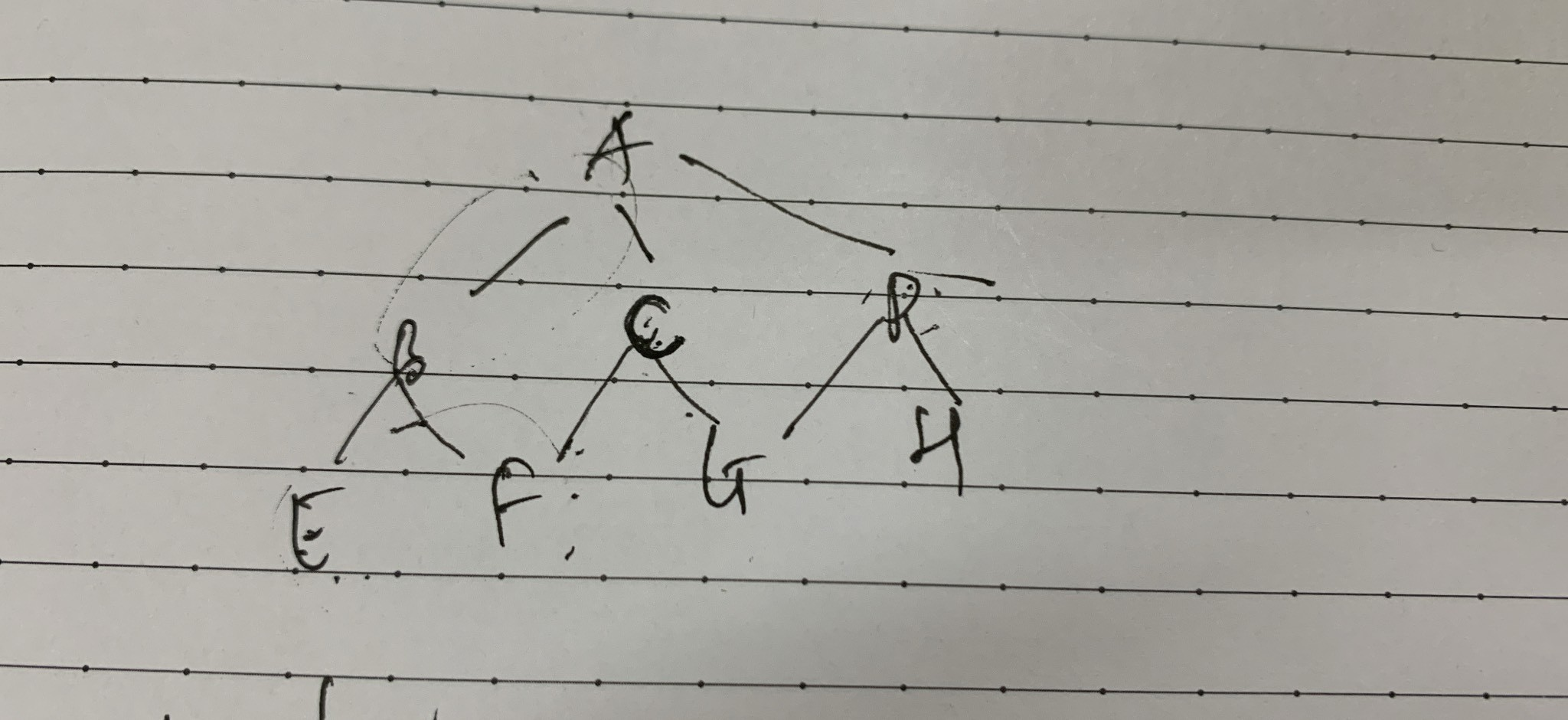
* 1. TOP-DOWN

|  |  |
| --- | --- |
|  | Stub |
| A-B | E,F,C,D |
| A-C | B,D,F,G |
| A-D | B,C,G,H |
| B-E | F,C,D |
| B-F | C,D |
| C-F | G,D |
| C-G | D |
| D-G | H |
| D-H | - |

656664

|  |  |
| --- | --- |
|  | stub |
| 1-2 | 3,4,5,6 |
| 1-3 | 2,4,7 |
| 1-4 | 2,3,8,9,10 |
| 2-5 | 11,12,6,3,4 |
| 2-6 | 5,7,4 |
| 3-7 | 5,4 |
| 4-8 | 5,9,10,13 |
| 4-9 | 5,8,10 |
| 4-10 | 5,13 |
| 5-11 | 12,13 |
| 5-12 | 13 |
| 5-13 |  |

* 1. Bottom up



|  |  |
| --- | --- |
|  | Driver |
| B-E | B,A |
| B-F | A |
| C-F | C,A |
| C-G | A |
| D-G | D,A |
| D-H | A |
| A-B | A |
| A-C | A |
| A-D | - |

* 1. Sandwich

|  |  |  |
| --- | --- | --- |
|  | Driver | Stub |
| B-E | B,A |  |
| B-F | A |  |
| C-F | C,A |  |
| C-G | A |  |
| D-G | D,A |  |
| D-H | A |  |
| A-B |  | C,D |
| A-C |  | D |
| A-D |  | - |

Câu 2.18.

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Condictions | Đi học >= 80% | Điểm giữa kì >0 | Điểm btl >0 | Điểm giữa kì = 10 | Điểm BTL =10 | Actions | Được thi | Không được thi | Miễn thi |
| Rule 1 | T | T | T | T | T |  | T |  | T |
| Rule 2 | T | F | T | - | - |  |  | T |  |
| Rule 3 | T | T | F | - | - |  |  | T |  |
| Rule 4 | T | F | F | - | - |  |  | T |  |
| Rule 5 | T | T | T | F | F |  | T |  |  |
| Rule 6 | T | T | T | T | F |  | T |  | T |
| Rule 7 | T | T | T | F | T |  | T |  | T |
| Rule 8 | F | - | - | - | - |  |  | T |  |

|  |  |  |
| --- | --- | --- |
| Test case | Input | Expected Result |
| 1 | Đi học >=80%, điểm GK=10, điểm BTL=10 | Miễn thi |
| 2 | Đi học >=80%, điểm GK<=0, điểm BTL>0 | Không được thi |
| 3 | Đi học >=80%, điểm GK <=0, điểm BTL>0 | Không được thi |
| 4 | Đi học >=80%, điểm GK <=0, điểm BTL<=0 | Không được thi |
| 5 | Đi học >=80%, điểm 10>GK>0, điểm 10>BTL>=0 | Được thi |
| 6 | Đi học >=80%, điểm GK =10, điểm 10>BTL>=0 | Miễn thi |
| 7 | Đi học >=80%, điểm 10>GK>0, điểm BTL=10 | Miễn thi |
| 8 | Đi học <80% | Không được thi |

Câu 2.22.

1. Biên

Gọi x là phần thu nhập cá nhân

Valid

+ Bậc thuế 1: 0<x<=60: Biên 60

+ Bậc thuê 2: 60<x<=120: Biên 61,120

+ Bậc thuế 3: 120 <x <=216: Biến 121, 216

+ Bậc thuế 4: 216 <x <=384: Biến 217, 384

+ Bậc thuế 5: 384<x <=624: Biến385, 624

+ Bậc thuế 6: 624<x <=960: Biến 625, 960

+ Bậc thuế 7: x>960: Biên 961

Invalid:

+ x<=0 : biên 0

|  |  |  |
| --- | --- | --- |
| TC | Input | expectedResult |
| 1 | 0 | Invalid |
|  | 60 | 60^\*0.05 |
|  | 61 |  |

1. Phân vùng tương đương

Valid:

+ đến 60: 0<x<=60

+ trên 60 đên 120: 60<x<=120

2.23.

b. lop tuong duong

* Kỳ hạn
  + Valid
    - Khong ky han
    - 7ngay
    - 14 ngay
    - ...
    - 24 thang
      * Invalid
        + Khong nằm trong danh sách tren
* Loai tien
  + Valid
    - VND
    - EUR
    - USD
      * Invalid
        + Loại tiền khác
* Số tiền gửi
  + Valid
    - >0
      * Invalid
        + <=0
* Số ngày thực tế
  + Valid:
    - Đúng kỳ hạn
    - Trước kỳ han
    - Sau kỳ hạn
      * Invalid
        + Số âm

Câu 3.1.

1. int binSearch(int x, int v[], int n){
2. int low, high, mid;
3. low = 0;
4. high = n - 1;
5. while (low<=high){
6. mid = (low + high)/2;
7. if (x<v[mid])
8. high = mid - 1;
9. else if (x > v[mid])
10. low = mid + 1;
11. else
12. return mid;
13. }
14. return -1;
15. }

Cau 3.19

. biên

Valid

+ 0< mức sử dụng nước <= 10 => Bien 1,10

+ 10< mức sử dụng nước <=20 => Biên 11,20

+ 20 < mức sử dụng nước <=30 => Biên 21,30

+ mức sử dụng nước > 30 => Biên 31

Invalid

+ mức sử dụng nước<=0 => Biên 0

Ta có bảng testcasse

|  |  |  |
| --- | --- | --- |
| Test case | Input | Expected result |
| 1 | Số đọc tháng này 100 số đọc tháng trước 100 | Invalid |
| 2 | Số đọc tháng này 100 số đọc tháng trước 99 | Cộng tiền hàng: 1\* 5973  Thuế suất: 1\*5973 \* 0.05  Phí bảo về mt: 1\*5973 \* 0.1  Tổng tiền: (1\* 5973)+( 1\*5973 \* 0.05)+( 1\*5973 \* 0.1) |
| 3 |  |  |