Họ và tên: Nguyễn Đặng Anh Phương

MSSV: 1813621

**TUTORIAL SOLUTION**

Question 1:

int mul ( Node root ) {

if (root.val == 0) return 0;

if ( root == null ) return 1 ;

else root.val ∗ mul (root.left)

∗ mul (root.right);

}

Question 2:

A. Console: 0210. Giải thích: Vì đây là cơ chế pass by value, nên giá trị được đưa vào sẽ được tạo 1 bản sao và mọi thay đổi chỉ có ý nghĩa trên bản sao này, do đó các giá trị ban đầu của i và a[i] không bị thay đổi.

B. Console: 2010. Giải thích: Vì đây là cơ chế pass by value-result, nên giá trị sau khi được thay đổi trên bản sao sẽ được viết lại vào in-output parameter (ở đây là i và a[i]). Vậy nên 2 giá trị in-output sau khi chạy swap sẽ bị thay đổi.

C. Console: 2010. Giải thích: Vì đây là cơ chế pass by ref, nên giá trị được đưa vào sẽ là địa chỉ của biến. Mọi thay đổi sẽ được xảy ra trực tiếp trên ref truyền vào.

D. Console: 2210. Giải thích: Vì đây là cơ chế pass by name, cho nên cách hoạt động gần giống pass by ref, khác ở chỗ giá trị của parameter được tính toán (evaluate) tại thời điểm nó được gọi. Cụ thể được trình bày như sau:

swap(i, a[i]); // giá trị truyền vào là i = 0 và a[i]

// Tại hàm swap: // x = i. y = a[i]

int t = x; // t = i = 0

x = y; // i = x = 2

y = t; // Lúc này, thực chất là gọi a[i], i = 2 nên a[i] = a[2], lệnh

// assignment làm cho a[2] = 0

Question 3:

A. Object bị thay đổi giá trị là bản sao của pointer P trỏ đến VECT A[1]. Giá trị của R^[1] lúc này bằng 17. Giải thích: do đây là cơ chế pass by value nên sẽ tạo 1 bản sao của parameter được truyền vào.

B. Object bị thay đổi là VECT B[1]. Giá trị của B[1] lúc này là 17. Giải thích: có thêm var là định nghĩa pass by ref của Pascal. Do S là thể hiện của Q, Q lại chứa địa chỉ của B nên B sẽ bị thay đổi dựa vào 2 biến trên.

C. R bị thay đổi. Nó chứa địa chỉ của B nếu thực thi thành công. Giải thích: do R là một object riêng không liên quan đến object nào nên nếu nó bị gán thì mọi biến khác không bị ảnh hưởng.

D. Q thay đổi. Lúc này nó chứa địa chỉ của bản sao của A được tạo khi gọi SUB2. Giải thích: do ở trong SUB2 S được pass by ref nên thay đổi trên S cũng có nghĩa thay đổi trên Q.

Question 4:

A. Số vòng lặp là vô hạn. Vì chỉ có expr2 được protect, nên n không thay đổi sau mỗi vòng. Còn i thì không có protect, nên vì trong vòng lặp i bị giảm 1 đơn vị nên sau vòng lặp dù được tăng 1 đơn vị nó vẫn duy trì giá trị cũ là 1.

B. Số vòng lặp là vô hạn. Vì index được protect nên nó chỉ tăng 1 đơn vị sau mỗi vòng lặp, n được tăng 1 đơn vị sau mỗi vòng lặp nên số vòng lặp là vô hạn.

C. Số vòng lặp là vô hạn. Vì i bị giảm 1 đơn vị và n được tăng 1 đơn vị sau mỗi vòng lặp.

D. Cả 2 được protect nên số vòng lặp sẽ là 5.