### Algorithm 2: Follow(PV P)

1. Best\_AF = P₁; // Khởi tạo PV tốt nhất là P₁  
2. for i = 1 to N do // Lặp qua tất cả các PV trong quần thể hiện tại  
3. Calculate dis = BitDiff(Pᵢ, P) using Eq. 11; // Tính khoảng cách bit khác biệt giữa Pᵢ và P  
4. if dis ≤ VD AND fitness(Pᵢ) > fitness(Best\_AF) then // Nếu khoảng cách nằm trong VD và fitness của Pᵢ tốt hơn  
5. Best\_AF = Pᵢ; // Cập nhật PV tốt nhất với Pᵢ  
6. end if  
7. end for  
8. dis' = BitDiff(Best\_AF, P); // Tính sự khác biệt giữa PV tốt nhất và PV hiện tại  
9. if (dis' > 0) then // Nếu có sự khác biệt giữa chúng  
10. is\_follow = true; // Đánh dấu rằng AF đang thực hiện hành vi "follow"  
11. Randomly generate a positive integer k no higher than dis'; // Sinh số nguyên dương k nhỏ hơn hoặc bằng dis'  
12. Update P by applying bitwise complement operation on k bits of it; // Cập nhật P bằng cách đảo bitwise trên k bit khác biệt  
13. if (I(P) ≥ min\_util AND IS(P) ∉ SHU) then // Nếu P đạt ngưỡng lợi ích tối thiểu và chưa thuộc SHU  
14. IS(P) → SHU; // Thêm P vào tập hợp SHU (các mục hữu ích cao)  
15. end if  
16. end if