

Câu 1. Tên bài: VALIA.CPP

Nhận xét: giá trị của vali phụ thuộc vào 2 yếu tố: có bao nhiêu vật đang được xét và trọng lượng của các vật. Do đó bảng phương án sẽ là bảng 2 chiều:

- $F(i,j)$: là tổng giá trị lớn nhất của vali khi xét từ vật 1, vật 2, ...vật i và trọng lượng của vali chưa vượt quá j . Chú ý rằng khi xét đến $F[i,j]$ thì các giá trị trên bảng phương án đều đã được tối ưu.

Tính $F(i,j)$: vật đang xét là A_i với trọng lượng của vali không được quá j . Có 2 khả năng xảy ra:

+ Nếu chọn A_i đưa vào vali, trọng lượng vali trước đó phải không quá $j-A_i$. Vì mỗi vật chỉ được chọn 1 lần nên giá trị lớn nhất của vali lúc đó là $F(i-1, j-A_i) + B_i$

+ Nếu không chọn A_i , giá trị của vali là như cũ (như lúc trước khi chọn A_i): $F(i-1, j)$

Tóm lại ta có: $F(i, j) = \max(F(i-1, j-A_i) + B_i, F(i-1, j))$.

Kết quả bài toán là $F(n, w)$

Độ phức tạp của thuật toán là $O(n*W)$.

Câu 2: Tên bài: VALIB.CPP

Lưu ý: Trong bài toán này, mỗi đồ vật có thể lựa chọn nhiều lần.

Ta cũng xây dựng các hàm và công thức giống như bài toán 1, chỉ có điều đặc biệt ở chỗ: Khi xét đến đồ vật i và có trọng lượng vali tối đa là j , ta thấy trường hợp nếu $A_i \leq j$ thì $F(i, j) = \max(F(i, j-A_i) + B_i, F(i-1, j))$. Bởi vì lúc này 1 đồ vật hoàn toàn có thể lấy nhiều lần, nên khi lấy 1 đồ vật thứ i thì hoàn toàn có khả năng đồ vật thứ i đã được lấy trước đó.

Kết quả bài toán là $f[n][w]$

Độ phức tạp $O(n * w)$.

Câu 3: Tên bài: SEQ.CPP

Ta chọn hàm mục tiêu là $F(i, j)$ có ý nghĩa là nếu có thể tạo ra tổng bằng j từ i phần tử đầu tiên của dãy thì $F(i, j) = true$, ngược lại $F(i, j) = false$.

Do đó, nếu $F(n, S) = true$ thì bài toán có nghiệm.

Ta có thể tính $F(i,j)$ theo công thức:

$$F(i, j) = true \text{ Nếu } F(i-1, j) = true \text{ hoặc } F(i-1, j-a[i]) = true.$$

Ta có thể nhận xét rằng để tính dòng thứ i , ta chỉ cần dòng $i-1$. Bảng phương án khi đó chỉ cần 1 mảng 1 chiều $F[0..S]$.

Độ phức tạp của thuật toán là $O(n*m)$ (với m là tổng của n số).

Câu 4. Tên bài: FARMER.CPP

Để thấy mảnh đất thứ i có A_i cây ôliu và dải đất thứ j có $B_j - 1$ cây ôliu. Ta xem các mảnh đất và dải đất là các “đồ vật”, đồ vật thứ k có khối lượng W_k và giá trị V_k (nếu k là mảnh đất i thì $W_k = V_k = A_i$, nếu k là dải đất j thì $W_k = B_j$, $V_k = B_j - 1$). Ta cần chọn các “đồ vật”, sao cho tổng “khối lượng” của chúng không vượt Q và tổng “giá trị” là lớn nhất. Đây chính là bài toán *VALIA* đã trình bày ở trên.

Độ phức tạp của thuật toán là $O(Q^*(m + n))$.

Câu 5. Tên bài: MONSTER.CPP

Gọi $F[i, j]$ là tổng độ hung dữ lớn nhất của các tên cướp mà Bờm mua chuộc được khi đi qua i tên cướp đầu tiên và dùng j đồng tiền vàng. Nếu Bờm không thể đi qua i tên cướp đầu tiên với j đồng vàng thì $F[i, j] = -1$.
- Các bài toán cơ sở:

$$F[1, j] = H[1] \text{ nếu } j \geq T[1]$$

$$\text{Ngược lại: } F[1, j] = -1.$$

Công thức truy hồi: ($i > 1$)

+ Bờm mua chuộc tên cướp thứ i thì :

$$F[i, j] = F[i - 1, j - T[i]] + H[i] \text{ (điều kiện: } F[i - 1, j - T[i]] \neq -1)$$

+ Bờm không mua chuộc tên cướp thứ i thì :

$$F[i, j] = F[i - 1, j] \text{ (điều kiện: } F[i - 1, j] \geq H[i])$$

Nghiệm của bài toán là j nhỏ nhất sao cho $F[n, j] \neq -1$ hoặc $F[n, v]$ nếu $F[n, v] \neq -1$

Độ phức tạp của thuật toán là $O(n * V)$.