Bài toán cái túi

1. Unbounded knapsack

* Nhiệm vụ Tìm các đồ vật để đặt trong một ba lô không bị chặn.

Các bước lập trình động

      (1) Xác định cấu trúc con tối ưu với các bài toán con trùng nhau.

      (2) Xác định một công thức đệ quy.

      (3) Sử dụng chương trình động để giải quyết vấn đề.

      (4) Nếu cần, theo dõi thông tin bổ sung để thuật toán từ (3) có thể giải quyết một vấn đề liên quan.

* Nhiệm vụ Tìm các đồ vật để đặt trong một ba lô không bị chặn.

(1) Xác định cấu trúc con tối ưu với các bài toán con trùng nhau.

      Câu lệnh vấn đề hạn chế chúng ta giảm số lượng mục.

      Bằng quá trình loại bỏ, chúng tôi lý do chúng ta phải giải quyết vấn đề cho ba lô nhỏ hơn.

* Nhiệm vụ Tìm các đồ vật để đặt trong một ba lô không bị chặn.

(1) Xác định cấu trúc con tối ưu với các bài toán con trùng nhau.

      Câu lệnh vấn đề hạn chế chúng ta giảm số lượng mục.

      Bằng quá trình loại bỏ, chúng tôi lý do chúng ta phải giải quyết vấn đề cho ba lô nhỏ hơn.

1. Knapsack

* Nhiệm vụ Tìm các đồ vật để bỏ vào một cái ba lô 0/1.

  Các bước lập trình động

      (1) Xác định cấu trúc con tối ưu với các bài toán con trùng nhau.

      (2) Xác định một công thức đệ quy.

      (3) Sử dụng chương trình động để giải quyết vấn đề.

      (4) Nếu cần, hãy theo dõi thông tin bổ sung để thuật toán từ (3) có thể giải quyết được một vấn đề liên quan.

* Nhiệm vụ Tìm các đồ vật để bỏ vào một cái ba lô 0/1.

  (1) Xác định cấu trúc con tối ưu với các bài toán con trùng nhau.

      Chúng ta có thể sử dụng cấu trúc con tối ưu giống như ba lô không?

* Nhiệm vụ Tìm các đồ vật để bỏ vào một cái ba lô 0/1.

  (1) Xác định cấu trúc con tối ưu với các bài toán con trùng nhau.

      Chúng ta có thể sử dụng cấu trúc con tối ưu giống như ba lô không?

      Không, bài toán con cần thông tin về các mục đã được sử dụng.

* Nhiệm vụ Tìm các đồ vật để bỏ vào một cái ba lô 0/1.

  (1) Xác định cấu trúc con tối ưu với các bài toán con trùng nhau.

      Chúng tôi lý giải rằng chúng ta phải giải quyết vấn đề cho một số lượng nhỏ hơn các mặt hàng và cho ba lô nhỏ hơn.

* Nhiệm vụ Tìm các đồ vật để bỏ vào một cái ba lô 0/1.

  (1) Xác định cấu trúc con tối ưu với các bài toán con trùng nhau.

      Xử lý các mặt hàng theo một cách tương tự như cách chúng tôi xử lý các đỉnh trong Floyd-Warshall; hạn chế tập hợp các mục được sử dụng cho một tập hợp cụ thể 0 đến j-1.

      Trường hợp 1 Nếu giải pháp tối ưu cho các mục j không sử dụng mục j.

* Nhiệm vụ Tìm các đồ vật để bỏ vào một cái ba lô 0/1.

  Các bước lập trình động

      (1) Xác định cấu trúc con tối ưu với các bài toán con trùng nhau.

      (2) Xác định một công thức đệ quy.

      (3) Sử dụng chương trình động để giải quyết vấn đề.

      (4) Nếu cần, hãy theo dõi thông tin bổ sung để thuật toán từ (3) có thể giải quyết được một vấn đề liên quan.

* Nhiệm vụ Tìm các đồ vật để đặt trong một chiếc ba lô không bị chặn.

  (2) Xác định một công thức đệ quy.

      Cho K [x, j] là giá trị tối ưu cho dung lượng x với mục j.

K [x, j] = { 0 nếu x hoặc j là 0

Hoặc max {K [x, j-1], K [x-wj, j-1] + vj} (khác)

* Nhiệm vụ Tìm các đồ vật để bỏ vào một cái ba lô 0/1.

  Các bước lập trình động

      (1) Xác định cấu trúc con tối ưu với các bài toán con trùng nhau.

      (2) Xác định một công thức đệ quy.

      (3) Sử dụng chương trình động để giải quyết vấn đề.

      (4) Nếu cần, hãy theo dõi thông tin bổ sung để thuật toán từ (3) có thể giải quyết được một vấn đề liên quan.