19.KNAPSACK

#include <stdio.h>

int max(int a, int b)

{

return (a > b) ? a : b;

}

int knapsack(int capacity, int weights[], int values[], int n) {

int dp[n + 1][capacity + 1];

for (int i = 0; i <= n; i++) {

for (int w = 0; w <= capacity; w++) {

if (i == 0 || w == 0)

dp[i][w] = 0;

else if (weights[i - 1] <= w)

dp[i][w] = max(values[i - 1] + dp[i - 1][w - weights[i - 1]], dp[i - 1][w]);

else

dp[i][w] = dp[i - 1][w];

}

}

return dp[n][capacity];

}

int main() {

int n;

printf("Enter the number of items: ");

scanf("%d", &n);

int weights[n], values[n];

printf("Enter the weights of the items:\n");

for (int i = 0; i < n; i++) {

scanf("%d", &weights[i]);

}

printf("Enter the values of the items:\n");

for (int i = 0; i < n; i++) {

scanf("%d", &values[i]);

}

int capacity;

printf("Enter the capacity of the knapsack: ");

scanf("%d", &capacity);

int maxValue = knapsack(capacity, weights, values, n);

printf("The maximum value that can be obtained is: %d\n", maxValue);

return 0;

}

OUTPUT

