#include <iostream>

#include <algorithm>

using namespace std;

struct Item {

double pr, wt, prPerWt;

bool selected; // Added to indicate whether the item is selected or not

int originalIndex; // Added to store the original index of the item

};

void bubbleSort(Item arr[], int n) {

for (int i = 0; i < n - 1; i++) {

for (int j = 0; j < n - i - 1; j++) {

if (arr[j].prPerWt < arr[j + 1].prPerWt) {

Item temp = arr[j];

arr[j] = arr[j + 1];

arr[j + 1] = temp;

}

}

}

}

double fractionalKnapsackGreedy(Item arr[], int n, int cap) {

double pr = 0;

bubbleSort(arr, n);

cout << "Items added to knapsack\nItem\tProfit\tWeight\tP/W\tTotalProfit\tCapacity\n";

int i = 0;

for (; i < n; i++) {

if (cap == 0) break;

if (arr[i].wt <= cap) {

cap -= arr[i].wt; // 3 things 1.capacity decrease 2. profit increase 3. selected array true

pr += arr[i].pr;

arr[i].selected = true;

} else {

pr += (cap \* arr[i].prPerWt);

cap = 0;

arr[i].selected = true;

cout << "Fraction\n"; // visaru nko

}

cout << arr[i].originalIndex << "\t" << arr[i].pr << "\t" << arr[i].wt << "\t" << arr[i].prPerWt << "\t\t" << pr << "\t" << cap << endl;

}

cout << "Total Items selected " << i << endl;

return pr;

}

int main() {

int n, cap;

cout << "Enter count of items and knapsack capacity >>>";

cin >> n >> cap;

Item arr[n];

cout << "Enter profits and weights\n";

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

Item item;

cin >> item.pr >> item.wt;

item.prPerWt = item.pr / item.wt;

item.selected = false;

item.originalIndex = i + 1; // Store the original index start from 1

arr[i] = item;

}

cout << endl;

double pr = fractionalKnapsackGreedy(arr, n, cap);

cout << "\nMaximum profit = " << pr << endl;

cout << "Selected Items: ";

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

if (arr[i].selected) {

cout << arr[i].originalIndex << " ";

}

}

cout << endl;

return 0;

}