如下代码的时间复杂度是哦o(n^2),因为其最大的循环是n^2,所以其时间复杂度就是o(n^2)。

import java.util.Scanner;

class Commodity{

int piece;//购买数量

int price;//购买价格

}

public class ZuiShaoFeiYongGouWu {

private static int MAXCODE = 999;//商品编码的最大值

private static int SALECOMB = 99;//优惠商品组合数

private static int KIND = 5; //商品种类

private static int QUANTITY = 5; //购买某种商品数量的最大值

private static int b;//购买商品种类数

private static int s;//当前优惠组合数

private static int[] num = new int[MAXCODE+1];//记录商品编码与商品种类的对应关系

private static int[] product = new int[KIND+1];//记录不同种类商品的购买数量

private static int[][] offer = new int[SALECOMB+1][KIND+1];//offer[i][j]: 商品组合的优惠价(j=0)；某种优惠组合中某种商品需要购买的数量(j>0)

private static Commodity[] purch = new Commodity[KIND+1];//记录不同商品的购买数量和购买价格

private static int[][][][][] cost = new int[QUANTITY+1][QUANTITY+1][QUANTITY+1][QUANTITY+1][QUANTITY+1];//记录本次购买的总花费

public static void main(String[] args){

init();

comp(1);

out();

}

private static void minicost(){

int i,j,k,m,n,p,minm;

minm = 0;

for(i=1; i<=b; i++)

minm += product[i]\*purch[i].price;

for(p=1; p<=s; p++){

i = product[1] - offer[p][1];

j = product[2] - offer[p][2];

k = product[3] - offer[p][3];

m = product[4] - offer[p][4];

n = product[5] - offer[p][5];

if(i>=0 && j>=0 && k>=0 && m>=0 && n>=0 && cost[i][j][k][m][n]+offer[p][0] < minm) minm = cost[i][j][k][m][n] + offer[p][0];

}

cost[product[1]][product[2]][product[3]][product[4]][product[5]] = minm;

}

private static void init(){

Scanner input = new Scanner(System.in);

int i,j,n,p,t,code;

for(i=0; i<100; i++)

for(j=0; j<6; j++)

offer[i][j] = 0;

for(i=0; i<6; i++){

purch[i] = new Commodity();

purch[i].piece = 0;

purch[i].price = 0;

product[i] = 0;

}

b = input.nextInt();

for(i=1; i<=b; i++){

code = input.nextInt();

purch[i].piece = input.nextInt();

purch[i].price = input.nextInt();

num[code] = i;

}

s = input.nextInt();

for(i=1; i<=s; i++){

t = input.nextInt();

for(j=1; j<=t; j++){

n = input.nextInt();

p = input.nextInt();

offer[i][num[n]] = p;

}

offer[i][0] = input.nextInt();

}

}

private static void comp(int i){

if(i > b){

minicost();

return;

}

for(int j=0; j<=purch[i].piece; j++){

product[i] = j;

comp(i+1);

}

}

private static void out(){

System.out.println(cost[product[1]][product[2]][product[3]][product[4]][product[5]]);

}

}