#include<iostream>

#include<vector>

using namespace std;

struct GoodsAttr

{int a;

int b;

int val;

}stGoodsAttr;

int max(int a,int b)

{

return(a>=b)?a:b;

}

int GetMaxvalue(vector<struct GoodsAttr>&godset,const int & v1,const int & v2,const int & k)

{

int i,j,m;

int temp=0;

vector<vector<int>>> value(v1+1,vector<vector<int>>(v2+1,vector<int>(k+1,0)));

for(auto&goods:goodset)

{

for(i=v1;i>=0;i--）

{

for(j=v2;j>=0;j--）

{

for(m=k;m>=0;m--）

{

if(m>=1)

{

temp=max(temp,value[i][j][m-1]+goods.val);

}

if(j>=goods.b)

{

temp=max(temp,value[i][j-goods.b][m]+goods.val);

}

if(i>=goods.a)

{

temp=max(temp,value[i-goods.b][j][m]+goods.val);

}

value[i][j][m]=max(temp,value[i][j][m]);

temp=0;

}

}

}

}

return value[v1][v2][k];

}

int main(int argc,char \*\*argv)

{

FILE \*fd;

vector<struct GoodsAttr> Goodset;

int n,v1,v2,k;

int maxval,tmp;

fd=fopen("C:\\Users\\Administrator\\Desktop\\test\\knapsack\\knapSackPrj\\goods.txt","r");

if(fd==nullptr)

{

cout<<"can't open the file!\n<<end1;

return 0;

}

cout<<"many dimensions knapsack algorithm\n<<end1;

while(!feof(fd))

{

fscanf(fd,"%d%d%d%d",&n,&v1,&v2,&k);

cout<<n<<""<<v1<<""<<v2<<""<<k<<end1;

tmp=n;

while(tmp--)

{

fscanf(fd,"%d%d%d%d"&stGoodsAttr.a,&sttGoodsAttr.b,&stGoodsAttr.va1);

cout<<stGoodsAttr.a<<""<<sttGoodsAttr.b<<""<<stGoodsAttr.va1<<end1;

Goodset.clear();

}

fclose(fd);

return 0;

}