**计算系数**

|  |
| --- |
| program factor;  //uses sysutils;  const      proname='factor';      yu=10007;  var      fin,fout:text;      i,j,k,m,n,l,r,s,t,x,y:longint;      a,b:int64;      ans:int64;      c:array[0..1100,0..1100]of longint;      procedure pin;  var      i,j:longint;  begin      readln(fin,a,b,k,n,m);  end;      function getc(n,r:int64):int64;  var      i,j:longint;  begin      fillchar(c,sizeof(c),0);      c[0,0]:=1;      c[1,0]:=1;      c[1,1]:=1;      for i:=2 to n do          begin          c[i,0]:=1;          for j:=1 to i do              c[i,j]:=(c[i-1,j]+c[i-1,j-1]) mod yu;          end;      exit(c[n,r] mod yu);  end;      procedure main;  var      i,j:longint;  begin      ans:=1;      a:=a mod yu;      b:=b mod yu;      for i:=1 to n do          ans:=ans\*a mod yu;      for i:=1 to m do          ans:=ans\*b mod yu;      if n>m then          t:=m      else          t:=n;      ans:=ans\*getc(k,t) mod yu;  end;      procedure pout;  var      i,j:longint;  begin      writeln(fout,ans);  end;      begin      assign(fin,proname+'.in');      assign(fout,proname+'.out');      reset(fin);      rewrite(fout);      //time:=now;      pin;      main;      pout;      //writeln(fout,(now-time)\*24\*3600\*1000:0:0);      close(fin);      close(fout);  end. |

**聪明的质监员**

|  |
| --- |
| program qc;  //uses sysutils;  const      proname='qc';  var      fin,fout:text;      i,j,k,m,n,l,r,t,x,y:longint;      w,v:array[0..200100]of longint;      a:array[0..200100,1..2]of longint;      sum:array[0..200100]of longint;      maxw,mid:longint;      now,s,ans:int64;      sumv:array[0..200100]of int64;      procedure pin;  var      i,j,k:longint;  begin      readln(fin,n,m,s);      fillchar(w,sizeof(w),0);      fillchar(v,sizeof(v),0);      fillchar(a,sizeof(a),0);      maxw:=0;      for i:=1 to n do          begin              readln(fin,w[i],v[i]);              if maxw<w[i] then                  maxw:=w[i];          end;      for i:=1 to m do          readln(fin,a[i,1],a[i,2]);  end;      function get(canshu:longint):int64;  var      i,j,k:longint;  begin      sum[0]:=0;      sumv[0]:=0;      for i:=1 to n do          begin              sum[i]:=sum[i-1];              sumv[i]:=sumv[i-1];              if w[i]>=canshu then                  begin                      sum[i]:=sum[i-1]+1;                      sumv[i]:=sumv[i-1]+v[i];                  end;          end;      get:=0;      for i:=1 to m do          get:=get+(sum[a[i,2]]-sum[a[i,1]-1])\*(sumv[a[i,2]]-sumv[a[i,1]-1]);  end;      function min(x,y:int64):int64;  begin      if x>y then exit(y) else exit(x);  end;    function max(x,y:int64):int64;  begin      if x>y then exit(x) else exit(y);  end;      procedure main;  var      i,j,k:longint;      t1,t2,t3:int64;  begin      l:=0;      r:=maxw;      ans:=10000000000000;      repeat          mid:=(l+r)shr 1;          now:=get(mid);          if ans>abs(now-s) then              ans:=abs(now-s);          if now<s then              r:=mid-1;          if now>s then              l:=mid+1;          if now=s then              exit;      until l>r;  end;      procedure pout;  var      i,j,k:longint;  begin      writeln(fout,ans);  end;      begin      assign(fin,proname+'.in');      assign(fout,proname+'.out');      reset(fin);      rewrite(fout);      //time:=now;      pin;      main;      pout;      //writeln(fout,(now-time)\*24\*3600\*1000:0:0);      close(fin);      close(fout);  end. |