```
> read("/home/ph/maple/RB.txt"):Delta:=evalm(1/2*(R-B)):A:=evalm(1/2*R+1/2*B):n:=rowdim(Delta):

> outfile:="/home/ph/maple/RBGap.txt";
  writeto(outfile);print();

> appendto(outfile);print(`R:=`);goMatGAP(R);print(`B:=`);goMatGAP(B);writeto(terminal):interface(echo=1):

> writeto(terminal):interface(echo=1):

> mr:=map(convert,matrans(R),decimal,hex);mb:=map(convert,matrans(B),decimal,hex);pd:=false:loop:=false:p
   prod1:=`*`(seq((mr[k]-mb[k]),k=1..n)):prod2:=`*`(seq((mr[k]-k),k=1..n)):prod3:=`*`(seq((k-mb[k]),k=1..n
  (not loop) then print("loopcheck",loop): fi;

> Delta:=evalm((R-B)/2):A:=evalm((R+B)/2):
  `R`=matrans(R);`B`=matrans(B);Omega:=abel(A):unassign('sigma'):liecliff(n):omega:=diag(seq(Omega[k,k],k

> print();if(rank(A)=rank(Delta)) then print("<--> CC <-->") else print("NOT CC",
  "RANKDIFF",rank(A)-rank(Delta)) fi;

> NN:=binomial(n,2):J2:=evalm(IdentityMatrix(NN)):uu:=vector(NN,1):unassign('t','e','x','tau'):e:=vector(n):phi:=

> dx:=rank(Delta):`rank of Delta`=dx,`for n equals`=n;`rank of
  A`=rank(A);NA:=NullSpace(Matrix(A)):NTA:=NullSpace(Matrix(transpose(A))):nu:=nullity(Delta):zeta:=vecto
  A",det(A),"nullspace",evalm(1/x[1]*linsolve(A,vector(n,0),'rr',x));"ker DELTA",nullspace(Delta);"A",factor(det(lam
  one-half",evalm(1/x[1]*linsolve(2*A-J,vector(n,0),'rr',x));"eigenvalue A minus
  one-half",evalm(1/x[1]*linsolve(2*A+J,vector(n,0),'rr',x));"Delta",factor(det(lambda*J-Delta));"eigenvalue Delta
  one-half",evalm(1/x[1]*linsolve(2*Delta-J,vector(n,0),'rr',x));"eigenvalue Delta minus one-half",evalm(1/x[1]*lin

> print();NSA:=nullspace(A);nullA:=nops(NSA):"DIM NULLSPACE OF
  A",nullA;#map(evalf,[eigenvals(A)]);map(evalf,[eigenvals(Delta)]);

> if(nullA=0) then print( "A IS INVERTIBLE") fi;

> AA:=sympow(A,2):D2:=sympow(Delta,2):

> nsj:=nullspace(evalm(J2-AA)):per:=evalb(nops(nsj)=0):print("is irred, aperiodic?",per):

    =====================================================================

> unassign('rho'):liecliff(n):

> ISX:=NullSpace(Matrix(transpose(evalm(Delta)))):ND:=NullSpace(Matrix(Delta)):

> CA:=r->if r>0 then concat(seq(NA[q],q=1..r))  else "" fi:CTA:=r->if r>0 then concat(seq(NTA[q],q=1..r))
   else "" fi:

> "ker A",CA(nullity(A)),"ker Tr A",CTA(nullity(A)),"ker Delta",concat(seq(ND[q],q=1..nullity(Delta))),"ker  tr
  Delta",concat(seq(ISX[q],q=1..nullity(Delta)));

> spi:=evalm(pi);pip:=add(pi[k]^2,k=1..n):printf("\t\t\t%s\t\t%6.2f","Upper Bound on r",1/pip):
```

$outfile := "/home/ph/maple/RBGap.txt"$

$mr := [3, 1, 1, 1, 2, 3]$

$mb := [5, 4, 4, 6, 4, 4]$

$prod := -622080$

$R = [3, 1, 1, 1, 2, 3]$

$B = [5, 4, 4, 6, 4, 4]$

$$\pi := \left[\frac{1}{4}, \frac{1}{16}, \frac{3}{16}, \frac{1}{4}, \frac{1}{8}, \frac{1}{8}\right]$$

$W := 16$

"<--> CC <-->"<--> CC <-->"" align="center">

$rank\ of\ Delta = 5,\ for\ n\ equals = 6$

$rank\ of\ A = 5$

$$\text{"Delta"}, \begin{bmatrix} 0 & 0 & 1 & 0 & -1 & 0 \\ 1 & 0 & 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & -1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & -1 \\ 0 & 1 & 0 & -1 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 \end{bmatrix}$$

"det A", 0, "nullspace", [1, 1, 1, -1, -1, -1]

"ker DELTA", {[1, 1, 1, 1, 1, 1]}

$$\text{"A"}, \frac{1}{2} \lambda^3 (\lambda - 1) (2 \lambda^2 + 2 \lambda + 1)$$

"eigenvalue A plus one-half", [0, 0, 0, 0, 0, 0]

"eigenvalue A minus one-half", [0, 0, 0, 0, 0, 0]

$$\text{"Delta"}, \frac{1}{2} \lambda^4 (-1 + 2 \lambda^2)$$

"eigenvalue Delta plus one-half", [0, 0, 0, 0, 0, 0]

"eigenvalue Delta minus one-half", [0, 0, 0, 0, 0, 0]

$NSA := \{[-1, -1, -1, 1, 1, 1]\}$

"DIM NULLSPACE OF A", 1

"is irred, aperiodic?", *true*

$$\text{"ker A"}, \begin{bmatrix} -1 \\ -1 \\ -1 \\ 1 \\ 1 \\ 1 \end{bmatrix}, \text{"ker Tr A"}, \begin{bmatrix} 0 \\ -1 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}, \text{"ker Delta"}, \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}, \text{"ker tr Delta"}, \begin{bmatrix} 0 \\ -1 \\ 1 \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

$$spi := \left[\frac{1}{4}, \frac{1}{16}, \frac{3}{16}, \frac{1}{4}, \frac{1}{8}, \frac{1}{8}\right]$$

Upper Bound on r 5.12

```
##################################### *RECOLOR HERE*
#############################################

> "ORIGINAL R",matrans(evalm(A+Delta));"ORIGINAL B",matrans(evalm(A-Delta));

> colrs:={4,5,6};

> ee:=vector(n,1):COLR:=colrs;for i to nops(COLR) do ee[convert(COLR[i],decimal,hex)]:=-1 od:

> unassign('COLR','e','f','t'):assume(t<=1 and t>=-1):

> phi:=diag(seq(ee[q],q=1..n)):Detta:=evalm(phi&*Delta):R:=evalm(A+Detta):B:=evalm(A-Detta):

> "CURRENT R",matrans(R);"CURRENT B",matrans(B);

> #######################################################

> rank(A),rank(Delta);print("NullSpace of
  "||Delta);nsd:=nullspace(Delta):seq(readVec(nsd[k]),k=1..nullity(Delta));

> NSA:=NullSpace(Matrix(A)): MSA:=NullSpace(Matrix(evalm(J-Detta))):IB:=IntersectionBasis([NSA,MSA]);

> ###

> #RUN2(R,B,1,false):

> ################## FROM R and B start HERE ###########

> liecliff(n):UJ:=matrix(n,n,1):J:=evalm(IdentityMatrix(n)):u:=vector(n,1):unassign('x'):
```

"ORIGINAL R", [3, 1, 1, 1, 2, 3]

"ORIGINAL B", [5, 4, 4, 6, 4, 4]

$colrs := \{4, 5, 6\}$

$COLR := \{4, 5, 6\}$

"CURRENT R", [3, 1, 1, 6, 4, 4]

"CURRENT B", [5, 4, 4, 1, 2, 3]

5, 5

"NullSpace of Delta"

{1, 2, 3, 4, 5, 6}

$$IB := \left\{ \begin{bmatrix} -1 \\ -1 \\ -1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \right\}$$

> unassign('s','t'):det(A);AT:=evalm(A+t*Detta);"det",det(AT);factor(%[2]);AT:=evalm(A-2/3*Detta):rank(pipow(

$$0$$

$$AT := \begin{bmatrix} 0 & 0 & \frac{1}{2}+\frac{1}{2}t & 0 & \frac{1}{2}-\frac{1}{2}t & 0 \\ \frac{1}{2}+\frac{1}{2}t & 0 & 0 & \frac{1}{2}-\frac{1}{2}t & 0 & 0 \\ \frac{1}{2}+\frac{1}{2}t & 0 & 0 & \frac{1}{2}-\frac{1}{2}t & 0 & 0 \\ \frac{1}{2}-\frac{1}{2}t & 0 & 0 & 0 & 0 & \frac{1}{2}+\frac{1}{2}t \\ 0 & \frac{1}{2}-\frac{1}{2}t & 0 & \frac{1}{2}+\frac{1}{2}t & 0 & 0 \\ 0 & 0 & \frac{1}{2}-\frac{1}{2}t & \frac{1}{2}+\frac{1}{2}t & 0 & 0 \end{bmatrix}$$

"det", 0

0

2, "vs", 5

> DD:=sympow(Detta,2):factor(det(J2-AA-s*DD));

$$-\frac{1}{16384}(4s^6+s^5-75s^4-94s^3-650s^2-1475s-2831)(s-1)^2$$

> A:=evalm(1/2*(R+B)):RBAR:=clos(R)[1]:BBAR:=clos(B)[1]:p:=1/2:

> print("RANK of R is ",rank(R));print("R ranking is ",rank(pipow(pi,R)),"vs",rank(R));

> X:=multiply(R,RBAR):"RBAR ranking",rank(pipow(pi,X)),"vs",rank(X);

> print("RANK of B is ",rank(B));print("B ranking is ",rank(pipow(pi,B)),"vs",rank(B));

> X:=multiply(B,BBAR):"BBAR ranking",rank(pipow(pi,X)),"vs",rank(X);

"RANK of R is ", 4

"R ranking is ", 1, "vs", 4

"RBAR ranking", 1, "vs", 4

"RANK of B is ", 5

"B ranking is ", 2, "vs", 5

"BBAR ranking", 1, "vs", 4

> CS1:=ColumnSpace(Matrix(R));CS2:=ColumnSpace(Matrix(B));

> IB:=IntersectionBasis([CS1,CS2]);print("Common Col Range has dimension",nops(IB));if(nops(IB)>0) then for i to nops(IB) do print(evalm(IB[i])); od;fi;

$$CS1 := \left[\begin{bmatrix}1\\0\\0\\0\\0\\0\end{bmatrix}, \begin{bmatrix}0\\1\\1\\0\\0\\0\end{bmatrix}, \begin{bmatrix}0\\0\\0\\1\\0\\0\end{bmatrix}, \begin{bmatrix}0\\0\\0\\0\\1\\1\end{bmatrix}\right]$$

$$CS2 := \left[\begin{bmatrix}1\\0\\0\\0\\0\\0\end{bmatrix}, \begin{bmatrix}0\\1\\1\\0\\0\\0\end{bmatrix}, \begin{bmatrix}0\\0\\0\\1\\0\\0\end{bmatrix}, \begin{bmatrix}0\\0\\0\\0\\1\\0\end{bmatrix}, \begin{bmatrix}0\\0\\0\\0\\0\\1\end{bmatrix}\right]$$

$$IB := \left[\begin{bmatrix}1\\0\\0\\0\\0\\0\end{bmatrix}, \begin{bmatrix}0\\1\\1\\0\\0\\0\end{bmatrix}, \begin{bmatrix}0\\0\\0\\1\\0\\0\end{bmatrix}, \begin{bmatrix}0\\0\\0\\0\\1\\1\end{bmatrix}\right]$$

"Common Col Range has dimension", 4

[1, 0, 0, 0, 0, 0]

[0, 1, 1, 0, 0, 0]

[0, 0, 0, 1, 0, 0]

[0, 0, 0, 0, 1, 1]

> CS1:=RowSpace(Matrix(R));CS2:=RowSpace(Matrix(B));

> IB:=IntersectionBasis([CS1,CS2]);print("Common Row Range has dimension",nops(IB));if(nops(IB)>0) then for i to nops(IB) do print(evalm(IB[i])); od;fi;

CS1 := [[1, 0, 0, 0, 0, 0], [0, 0, 1, 0, 0, 0], [0, 0, 0, 1, 0, 0], [0, 0, 0, 0, 0, 1]]

CS2 := [[1, 0, 0, 0, 0, 0], [0, 1, 0, 0, 0, 0], [0, 0, 1, 0, 0, 0], [0, 0, 0, 1, 0, 0], [0, 0, 0, 0, 1, 0]]

IB := [[0, 0, 1, 0, 0, 0], [0, 0, 0, 1, 0, 0], [1, 0, 0, 0, 0, 0]]

"Common Row Range has dimension", 3

[0, 0, 1, 0, 0, 0]

[0, 0, 0, 1, 0, 0]

[1, 0, 0, 0, 0, 0]

> nullspace(R);nullspace(B);

> nullspace(transpose(R));nullspace(transpose(B));

{[0, 1, 0, 0, 0, 0], [0, 0, 0, 0, 1, 0]}

{[0, 0, 0, 0, 0, 1]}

{[0, -1, 1, 0, 0, 0], [0, 0, 0, 0, -1, 1]}

{[0, -1, 1, 0, 0, 0]}

```
> p:=1/2:

> A2:=evalm(p*sympow(R,2)+(1-p)*sympow(B,2)):N2:=binomial(n,2):J2:=IdentityMatrix(N2):

> dim:=nops(nullspace(J2-A2));uu:=vector(N2,1):ab:=false:if(dim>1) then ab:=true;AB:=abel(A2) fi:

> MB:=clearDenoms(evalm(linsolve(J2-transpose(A2),vector(N2,0),'rr',x)));

> unassign('MBASIS'):for i to dim do MBASIS[i]:=matvec(map(diff,MB,x[i])); od;

> NB:=clearDenoms(evalm(linsolve(J2-A2,vector(N2,0),'rr',x)));

> unassign('NBASIS'):for i to dim do NBASIS[i]:=matvec(map(diff,NB,x[i])); od;

> "CC characteristic",evalm(1/2*(UJ-symult(phi,UJ)));

> if(dim>1) then u2:=evalm(AB&*uu) ;pi2:=evalm(uu&*AB) ;print("FIXED POINTS DIM ",dim) else

> pi2:=map(simplify,evalm(1/x[1]*MB)); u2:=map(simplify,evalm(1/x[1]*NB)); fi;

> M:=matvec(clearDenoms(pi2)):

> m:=max(seq(u2[k],k=1..N2)):

> NX:=matvec(u2):rh:=multiply(spi,NX):rk:=simplify(m/(m-rh[1])):pp:=multiply(NX,u/m):N:=evalm(1/m*NX):

> if(iszero(M)) then rk:=1 fi;

> print("RANK of N is ",rank(N),"RANK of M is ",rank(M));print("RANK of the KERNEL is ",rk);

> outfile:="/home/ph/maple/RBGap.txt";
  writeto(outfile);print();

> appendto(outfile);print(`R:=`);goMatGAP(R);print(`B:=`);goMatGAP(B);writeto(terminal):interface(echo=1):
```

$$dim := 2$$

$$MB := [x_1, 3\ x_1, 4\ x_2, 2\ x_2, 2\ x_2, 0, x_2, x_2, 0, 3\ x_2, x_2, 2\ x_2, 2\ x_1, 2\ x_1, 0]$$

$$MBASIS_1 := \begin{bmatrix} 0 & 1 & 3 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 3 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & 2 \\ 0 & 0 & 0 & 2 & 0 & 0 \\ 0 & 0 & 0 & 2 & 0 & 0 \end{bmatrix}$$

$$MBASIS_2 := \begin{bmatrix} 0 & 0 & 0 & 4 & 2 & 2 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 3 & 1 & 2 \\ 4 & 1 & 3 & 0 & 0 & 0 \\ 2 & 1 & 1 & 0 & 0 & 0 \\ 2 & 0 & 2 & 0 & 0 & 0 \end{bmatrix}$$

$$NB := [x_1, x_1, x_2, x_2, x_2, 0, x_2, x_2, x_2, x_2, x_2, x_2, x_1, x_1, 0]$$

$$NBASIS_1 := \begin{bmatrix} 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix}$$

$$NBASIS_2 := \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 \end{bmatrix}$$

$$\text{"CC characteristic"}, \begin{bmatrix} 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 1 & 0 & 0 & 0 \end{bmatrix}$$

$$u2 := [1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 0]$$

$$\pi 2 := \left[\frac{1}{2}, \frac{3}{2}, \frac{9}{4}, \frac{9}{8}, \frac{9}{8}, 0, \frac{9}{16}, \frac{9}{16}, 0, \frac{27}{16}, \frac{9}{16}, \frac{9}{8}, 1, 1, 0\right]$$

"FIXED POINTS DIM ", 2

$$\text{"PROTO-M"}, \begin{bmatrix} 0 & 8 & 24 & 36 & 18 & 18 \\ 8 & 0 & 0 & 9 & 9 & 0 \\ 24 & 0 & 0 & 27 & 9 & 18 \\ 36 & 9 & 27 & 0 & 16 & 16 \\ 18 & 9 & 9 & 16 & 0 & 0 \\ 18 & 0 & 18 & 16 & 0 & 0 \end{bmatrix}, \text{"N"}, \begin{bmatrix} 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 \end{bmatrix}$$

"RANK of N is ", 4, "RANK of M is ", 6

"RANK of the KERNEL is ", 4

outfile := "/home/ph/maple/RBGap.txt"

> clearDenoms(pi);clearDenoms(pi2);

[4, 1, 3, 4, 2, 2]

```
[8, 24, 36, 18, 18, 0, 9, 9, 0, 27, 9, 18, 16, 16, 0]
```

```
> unassign('x'):
```

```
> RK:=sympow(R,rk):BK:=sympow(B,rk):NK:=binomial(n,rk):JK:=evalm(IdentityMatrix(NK)):AK:=evalm((1/2)*
```

```
> pi||rk:=clearDenoms(evalm(1/x[1]*(linsolve(transpose(JK-AK),vector(NK,0),'rr',x)))):u||rk:=clearDenoms(eva
```

```
> `<br/>KERNEL RANK IS `;rk;`<p/>`;print("pi"||rk,pi||rk);`<p/>`;"supp
  pi"||rk,readVec(pi||rk);`<p/>`;print("u"||rk,u||rk);`<p/>`;
```

```
> "supp u"||rk,readVec(u||rk);`<br/>` ;
```

```
> unassign('BETA','RRG'):nr:=nops(readVec(pi||rk)):CK:=map(convert,choose(n,rk),set):print(`<p/>`);
```

```
> for i to nr do
  BETA[i]:=[CK[readVec(pi||rk)[i]],pi||rk[readVec(pi||rk)[i]]];RRG[i]:=BETA[i][1];print("Range",i,RRG[i])
  od:
```

```
> GPN:=[goParts(N)]:unassign('PPT');
```

```
> gp:=getParts(GPN,n,rk,BETA[1][1],R,B,N):for i to nops(gp[1]) do
  PPT[i]:=readcycles(gp[1][i]);od;alpha:=gp[2];RPARTS:=gp[3];BPARTS:=gp[4];beta:=vector([seq(BETA[k][2],
```

```
<br/>KERNEL RANK IS

4

<p/>

"pi4", [0, 0, 0, 1, 0, 0, 1, 2, 0, 0, 0, 0, 0, 0, 0]

<p/>

"supp pi4", {4, 7, 8}

<p/>

"u4", [0, 0, 0, 1, 1, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0]

<p/>

"supp u4", {4, 5, 7, 8}

<br/>

<p/>

"Range", 1, {1, 2, 4, 5}

"Range", 2, {1, 3, 4, 5}

"Range", 3, {1, 3, 4, 6}
```

PPT_1 := {{1}, {2, 3}, {5, 6}, {4}}

α := [1]

$$RPARTS := []$$

$$BPARTS := []$$

$$\beta := \left[\frac{1}{4}, \frac{1}{4}, \frac{1}{2}\right]$$

```
> XC:={}:print();for i to nops(gp[1]) do

          rcc:=PPT[i];
          XC:=XC union convert(rcc,set);

          printf(cat("\t\t\t\t",`alpha(`,convert([op(rcc)],string),`) =
  `,numer(alpha[i]),`/`,denom(alpha[i]),"\n\n"));
      od:

          npp:=nops(XC);

          unassign('XRC');print();

          APARTS:=matrix(npp,npp,0):
          RPARTS:=matrix(npp,npp,0):
          BPARTS:=matrix(npp,npp,0):print();

      for ky to npp do
          XRC[ky]:=vector(n,i->if(member(i,XC[ky])) then 1 else 0 fi):
          printf(cat("\t\t\t\t\t",`b`,ky,` = `,convert(XC[ky],string),"\n\n"));
      od:

       for iy to npp do
              aa:=multiply(R,XRC[iy]):
              bb:=multiply(B,XRC[iy]):
          for jy to npp do
              if(iszero(evalm(XRC[jy])-aa)) then
                  RPARTS[iy,jy]:=RPARTS[iy,jy]+1
              fi:
              if( iszero(XRC[jy]-bb)) then
                  BPARTS[iy,jy]:=BPARTS[iy,jy]+1
              fi:
          od:
      od:

       APARTS:=evalm(RPARTS+BPARTS);
       print(`Action of R and B on the blocks of the partitions:`);
      printf("\t\t\t\t R-PARTS, ");
          print(matrans(RPARTS));
      printf("\t\t\t\t B-PARTS, ");
          print(matrans(BPARTS));
      print();
          jj:=evalm(IdentityMatrix(npp)):
          nh:=nullspace(2*jj-transpose(APARTS)):
          nhh:=nh[1]:
          print(`with invariant measure`,clearDenoms(nhh));
          ptn:=evalm(add(nhh[k]*sym(XRC[k]),k=1..npp));
          print(`N by blocks,`);
          ptn:=evalm(ptn/ptn[1,1]);
          printf(`\t\t\t\t\t N - check: `);print( iszero(evalm(UJ-ptn)-N));
```

alpha([{1}, {2, 3}, {5, 6}, {4}]) = 1/1

$$npp := 4$$

b1 = {1}

b2 = {2, 3}

b3 = {5, 6}

b4 = {4}

$$APARTS := \begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 0 \end{bmatrix}$$

Action of R and B on the blocks of the partitions:

R-PARTS,

[2, 1, 4, 3]

B-PARTS,

[4, 3, 1, 2]

with invariant measure, [1, 1, 1, 1]

$$ptn := \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}$$

N by blocks,

$$ptn := \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}$$

N - check:

true

> "Partition Vectors";PV:=concat(seq(XRC[k],k=1..npp));

"Partition Vectors"

$$PV := \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

```
> NS:=nullspace(N):NXN:=stackmatrix(seq(NS[k],k=1..nullity(N)));multiply(NXN,s*R+t*B);iszero(multiply(%,N)
```

$$NXN := \begin{bmatrix} 0 & -1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -1 & 1 \end{bmatrix}$$

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & -t & t & 0 & 0 & 0 \end{bmatrix}$$

$$true, true$$

```
> rank(PV),rank(UJ-N),rank(N);
```

$$4, 4, 4$$

```
> H:=Centr(R,B);iszero(cmm(N,H));
```

$$H := 0$$

Error, (in linalg:-iszero) argument must be a matrix or a vector

****

CHECK IDEMS HERE.

****

```
> unassign('KBAS','IDEMS','Q','rg','prt');

    GK:=goKernel(R,B,rk):

    Q:=GK[1];
    IDEMS:=GK[2];
    KBAS:=GK[3];

    ranges:={}:
    partitions:={}:

    for i to doDIM(IDEMS) do
        rg[i]:={seq(readcycles(IDEMS[i])[k][1],k=1..rk)};
        ranges:=ranges union {rg[i]}:
        prt[i]:=readcycles(symult(IDEMS[i],J));
        partitions:=partitions union {prt[i]}:
    od:

    nr:=nops(ranges);
    np:=nops(partitions);

    unassign('P','PT','PP'):# NUMBER OF PARTITIONS is np
    unassign('RG'):# NUMBER OF RANGES is nr

    idx:=vector(np):
    ct:=0:

    for i to doDIM(IDEMS) do
        if(rg[i]=rg[1]) then
            ct:=ct+1:
            idx[ct]:=i
        fi;
    od;
```

```
for i to np do PT||i:=prt[idx[i]]; od:

idx:=vector(nr):
ct:=0:

for i to doDIM(IDEMS) do
        if(prt[i]=prt[1]) then
                ct:=ct+1:
                idx[ct]:=i
        fi;
od;

for i to nr do
        RG||i:=rg[idx[i]];
        print(`<p/>`);
od:

for i to np do
        print("PT"||i=PT||i);
        print(`<p/>`);
od;

for i to nr do
        print("RG"||i=RG||i);
        print(`<p/>`);
od;

unassign('E','C','qprt','beta','qrg','Q2'):

GRPSIZE:=doDIM(Q)/nr/np:BASEGRP:=[]:

for i to doDIM(Q) do

qrg[i]:={seq(readcycles(Q[i])[k][1],k=1..rk)};
qprt[i]:=readcycles(symult(Q[i],J)):

        if(qprt[i]=PT1 and qrg[i]=RG1) then
                BASEGRP:=[op(BASEGRP),Q[i]]
        fi;
od:

for i to np do
for j to nr do
        E||i||j:=makeIdem(RG||j,PT||i):
        C:=Matrix(E||i||j):ECOL:=[]:
        for ix to n do
                if( not iszero(Column(C,ix))) then
                        ECOL:=[op(ECOL),ix]
                fi;
        od:
        C:=concat(seq(Column(C,ix),ix in ECOL)):
        FCOL:={seq(ix,ix=1..n)} minus {op(ECOL)}:
        C||i||j:=C:
        for ix in FCOL do
                C||i||j:=concat(C||i||j,UnitVector(ix,n))
        od:
od:
od:
```

$$Q := \left[\begin{bmatrix}0&0&1&0&0&0\\1&0&0&0&0&0\\1&0&0&0&0&0\\0&0&0&0&0&1\\0&0&0&1&0&0\\0&0&0&1&0&0\end{bmatrix}, \begin{bmatrix}1&0&0&0&0&0\\0&0&1&0&0&0\\0&0&1&0&0&0\\0&0&0&1&0&0\\0&0&0&0&0&1\\0&0&0&0&0&1\end{bmatrix}, \begin{bmatrix}0&0&0&1&0&0\\0&0&0&0&0&1\\0&0&0&0&0&1\\0&0&1&0&0&0\\1&0&0&0&0&0\\1&0&0&0&0&0\end{bmatrix}, \begin{bmatrix}0&0&0&1&0&0\\0&0&0&0&1&0\\0&0&0&0&1&0\\0&0&1&0&0&0\\1&0&0&0&0&0\\1&0&0&0&0&0\end{bmatrix}, \begin{bmatrix}0&1&0&0&0&0\\1&0&0&0&0&0\\1&0&0&0&0&0\\0&0&0&0&1&0\\0&0&0&1&0&0\\0&0&0&1&0&0\end{bmatrix},\right.$$

$$\begin{bmatrix}0&0&0&0&0&1\\0&0&0&1&0&0\\0&0&0&1&0&0\\1&0&0&0&0&0\\0&0&1&0&0&0\\0&0&1&0&0&0\end{bmatrix}, \begin{bmatrix}0&0&0&0&1&0\\0&0&0&1&0&0\\0&0&0&1&0&0\\1&0&0&0&0&0\\0&0&1&0&0&0\\0&0&1&0&0&0\end{bmatrix}, \begin{bmatrix}1&0&0&0&0&0\\0&0&1&0&0&0\\0&0&1&0&0&0\\0&0&0&1&0&0\\0&0&0&0&1&0\\0&0&0&0&1&0\end{bmatrix}, \begin{bmatrix}1&0&0&0&0&0\\0&1&0&0&0&0\\0&1&0&0&0&0\\0&0&0&1&0&0\\0&0&0&0&1&0\\0&0&0&0&1&0\end{bmatrix}, \begin{bmatrix}0&0&0&1&0&0\\0&0&0&0&1&0\\0&0&0&0&1&0\\0&1&0&0&0&0\\1&0&0&0&0&0\\1&0&0&0&0&0\end{bmatrix},$$

$$\left.\begin{bmatrix}0&0&1&0&0&0\\1&0&0&0&0&0\\1&0&0&0&0&0\\0&0&0&0&1&0\\0&0&0&1&0&0\\0&0&0&1&0&0\end{bmatrix}, \begin{bmatrix}0&0&0&0&1&0\\0&0&0&1&0&0\\0&0&0&1&0&0\\1&0&0&0&0&0\\0&1&0&0&0&0\\0&1&0&0&0&0\end{bmatrix}\right]$$

$$IDEMS := \left[\begin{bmatrix}1&0&0&0&0&0\\0&0&1&0&0&0\\0&0&1&0&0&0\\0&0&0&1&0&0\\0&0&0&0&0&1\\0&0&0&0&0&1\end{bmatrix}, \begin{bmatrix}1&0&0&0&0&0\\0&0&1&0&0&0\\0&0&1&0&0&0\\0&0&0&1&0&0\\0&0&0&0&1&0\\0&0&0&0&1&0\end{bmatrix}, \begin{bmatrix}1&0&0&0&0&0\\0&1&0&0&0&0\\0&1&0&0&0&0\\0&0&0&1&0&0\\0&0&0&0&1&0\\0&0&0&0&1&0\end{bmatrix}\right]$$

$$KBAS := [KBASIS_1, KBASIS_2, KBASIS_3, KBASIS_4, KBASIS_5, KBASIS_6, KBASIS_7, KBASIS_8, KBASIS_9, KBASIS_{10}, KBASIS_{11}, KBASIS_{12}]$$

$nr := 3$

$np := 1$

<p/>

<p/>

<p/>

"PT1" = {{1}, {2, 3}, {5, 6}, {4}}

<p/>

"RG1" = {1, 3, 4, 6}

<p/>

"RG2" = {1, 3, 4, 5}

<p/>

"RG3" = {1, 2, 4, 5}

<p/>

```
> #ssystem("/usr/local/bin/KBAS "|| rk);

> nulN:=nullity(NX):print(`<p/>`);
   rnkN:=rank(NX);print(`<p/>`);
  nsn:=nullspace(NX):

  if(rnkN<n) then
      NSN:=stackmatrix(seq(nsn[k],k=1..nulN)):

   unassign('y'):

      VRN:=linsolve(NSN,vector(nulN,0),'rr',y):

   for i to rnkN do
      idx||i:={};
      vv:=map(diff,VRN,y[i]);
          for j to n do
              if(vv[j]=1) then
                  idx||i:={op(idx||i),j}
              elif (vv[j]=-1) then
                  idx||i:={op(idx||i),[j]}
      fi
          od:
              od:
      print(`<p/>`);print(`Range of N`, VRN);
      print(`<br/>`);
      print(`Partitions`,seq(idx||k,k=1..rnkN));
  fi:

> print(`<p/>`);"Rank A",rank(A),"Rank Delta",rank(Delta);print(`<p/>`);

> LL:=linsolve(A,vector(n,0),'rr',lambda):"Nullspace of
  A",evalm(LL);print(`<p/>`);LL:=linsolve(Delta,vector(n,0),'rr',mu):"Nullspace of Delta",evalm(LL);
```

<p/>

$rnkN := 4$

<p/>

<p/>

Range of N, $[y_4, y_3, y_3, y_2, y_1, y_1]$

Partitions, {5, 6}, {4}, {2, 3}, {1}

<p/>

"Rank A", 5, "Rank Delta", 5

<p/>

"Nullspace of A", $[\lambda_1, \lambda_1, \lambda_1, -\lambda_1, -\lambda_1, -\lambda_1]$

<p/>

"Nullspace of Delta", $[\mu_1, \mu_1, \mu_1, \mu_1, \mu_1, \mu_1]$

```
> "GROUP HAS ORDER",doDIM(BASEGRP);

> "g",evalm(1/GRPSIZE*add(BASEGRP[k],k=1..nops(BASEGRP))),"gg*",evalm(1/GRPSIZE*add(symult(BASEGRP[k

> MAVG:=evalm(1/GRPSIZE*add(symmult(BASEGRP[k],UJ),k=1..nops(BASEGRP))):

> "g*g",evalm(1/GRPSIZE*add(symmult(BASEGRP[k],J),k=1..nops(BASEGRP))),"g*Jg",evalm(MAVG);

> "kernel has",doDIM(Q),"elements";
```

"GROUP HAS ORDER", 4

$$\text{"g"}, \begin{bmatrix} \frac{1}{4} & 0 & \frac{1}{4} & \frac{1}{4} & 0 & \frac{1}{4} \\ \frac{1}{4} & 0 & \frac{1}{4} & \frac{1}{4} & 0 & \frac{1}{4} \\ \frac{1}{4} & 0 & \frac{1}{4} & \frac{1}{4} & 0 & \frac{1}{4} \\ \frac{1}{4} & 0 & \frac{1}{4} & \frac{1}{4} & 0 & \frac{1}{4} \\ \frac{1}{4} & 0 & \frac{1}{4} & \frac{1}{4} & 0 & \frac{1}{4} \\ \frac{1}{4} & 0 & \frac{1}{4} & \frac{1}{4} & 0 & \frac{1}{4} \end{bmatrix}, \text{"gg*"}, \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 1 & 1 \end{bmatrix}$$

$$\text{"g*g"}, \begin{bmatrix} \frac{3}{2} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & \frac{3}{2} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{3}{2} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{3}{2} \end{bmatrix}, \text{"g*Jg"}, \begin{bmatrix} \frac{5}{2} & 0 & 2 & \frac{9}{4} & 0 & \frac{9}{4} \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 2 & 0 & \frac{5}{2} & \frac{9}{4} & 0 & \frac{9}{4} \\ \frac{9}{4} & 0 & \frac{9}{4} & \frac{5}{2} & 0 & 2 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ \frac{9}{4} & 0 & \frac{9}{4} & 2 & 0 & \frac{5}{2} \end{bmatrix}$$

"kernel has", 12, "elements"

```
> ABR:=abel(R):ABB:=abel(B):print("R-lim",ABR,"B-lim",ABB);

> "abel(R) in Vec(K)?",readcycles(ABR),checkspan(KBAS,ABR);"abel(B) in
  Vec(K)?",readcycles(ABB),checkspan(KBAS,ABB);
```

$$\text{"R-lim"}, \begin{bmatrix} \frac{1}{2} & 0 & \frac{1}{2} & 0 & 0 & 0 \\ \frac{1}{2} & 0 & \frac{1}{2} & 0 & 0 & 0 \\ \frac{1}{2} & 0 & \frac{1}{2} & 0 & 0 & 0 \\ 0 & 0 & 0 & \frac{1}{2} & 0 & \frac{1}{2} \\ 0 & 0 & 0 & \frac{1}{2} & 0 & \frac{1}{2} \\ 0 & 0 & 0 & \frac{1}{2} & 0 & \frac{1}{2} \end{bmatrix}, \text{"B-lim"}, \begin{bmatrix} \frac{1}{4} & \frac{1}{4} & 0 & \frac{1}{4} & \frac{1}{4} & 0 \\ \frac{1}{4} & \frac{1}{4} & 0 & \frac{1}{4} & \frac{1}{4} & 0 \\ \frac{1}{4} & \frac{1}{4} & 0 & \frac{1}{4} & \frac{1}{4} & 0 \\ \frac{1}{4} & \frac{1}{4} & 0 & \frac{1}{4} & \frac{1}{4} & 0 \\ \frac{1}{4} & \frac{1}{4} & 0 & \frac{1}{4} & \frac{1}{4} & 0 \\ \frac{1}{4} & \frac{1}{4} & 0 & \frac{1}{4} & \frac{1}{4} & 0 \end{bmatrix}$$

"abel(R) in Vec(K)?", {{1, 3}, {4, 6}}, *true*, $\left[\frac{1}{2}, 0, \frac{1}{2}, 0, 0, 0, 0, 0, 0, 0, 0, 0\right]$

"abel(B) in Vec(K)?", {{1, 2, 4, 5}}, *true*, $\left[\frac{1}{4}, \frac{1}{4}, 0, \frac{1}{4}, \frac{1}{4}, 0, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}\right]$

```
> goRange(ABR);goRange(ABB);goRange(N);#sym(ABR),sym(ABB);
```

$[y_1, 0, y_1, y_2, 0, y_2]$

$[y_1, y_1, 0, y_1, y_1, 0]$

$[y_1, y_2, y_2, y_3, y_4, y_4]$

```
> Qstack:=stackmatrix(seq(convert(evalm(Q[i]),vector),i=1..doDIM(Q))):rowdim(Qstack);vb:=RowSpace(Matrix(

> rankK:=rank(Qstack);

> kernull:=nullspace(Qstack):Kernull:=matrix(n,n,evalm(add(a[k]*kernull[k],k=1..nops(kernull))));
```

12

rankK := 12

$Kernull := [[a_{15}, a_{22}, a_{22}, a_2, a_{19}, a_{19}], [a_8, a_5, a_{14}, a_{16}, -a_2 - a_9 - a_{12} - a_{21} - a_{24}, -a_2 - a_3 - a_9 - a_{12} - a_{21}],$
$[-a_8 - a_{11} - a_{17} - a_{18} - a_{22}, -a_4 - a_5 - a_7 - a_{13} - a_{15}, -a_4 - a_7 - a_{13} - a_{14} - a_{15}, -a_6 - a_{10} - a_{16} - a_{19} - a_{20}, a_{24}, a_3],$
$[a_{10}, a_9, a_9, a_7, a_{11}, a_{11}], [a_{12}, a_6 + a_{20} - a_{23}, a_{20}, a_{18}, -a_1 + a_4 + a_{13}, a_4], [a_{21}, a_{23}, a_6, a_{17}, a_1, a_{13}]]$

COLLECT LOCAL GROUPS

```
> unassign('QV','QQ');for i to np do for j to nr do QV[i,j]:=[] od:od:

> for i to doDIM(Q) do for k to nr do if(qrg[i]=RG||k) then ix2:=k;fi;od; for k to np do if(qprt[i]=PT||k) then
  ix1:=k  fi;od; QV[ix1,ix2]:=[op(QV[ix1,ix2]),matrans(Q[i])]; od:

> for i to np do for j to nr do if(nops(QV[i,j][1])=1) then QV[i,j]:=op(3..nops(QV[i,j]),QV[i,j]);fi;
  od:od:grp:=nops(QV[1,1]);

> for i to np do print(i,"partition",PT||i); for j to nr do print(j,"range",convert(RG||j,list),QV[i,j]) ;

> od:od:

> for jx to grp do QQ[jx]:=submatrix(multiply(inverse(C11),transmat(QV[1,1][jx]),C11),1..rk,1..rk) ;od:
```

grp := 4

1, "partition", {{1}, {2, 3}, {5, 6}, {4}}

1, "range", [1, 3, 4, 6], [[*3, 1, 1, 6, 4, 4*], [*1, 3, 3, 4, 6, 6*], [*4, 6, 6, 3, 1, 1*], [*6, 4, 4, 1, 3, 3*]]

2, "range", [1, 3, 4, 5], [[*4, 5, 5, 3, 1, 1*], [*5, 4, 4, 1, 3, 3*], [*1, 3, 3, 4, 5, 5*], [*3, 1, 1, 5, 4, 4*]]

3, "range", [1, 2, 4, 5], [[*2, 1, 1, 5, 4, 4*], [*1, 2, 2, 4, 5, 5*], [*4, 5, 5, 2, 1, 1*], [*5, 4, 4, 1, 2, 2*]]

```
> unassign('x','y','z','t'):F1:=matrix(n,n,0):TT:=matrix(np,nr,0):

> for i to np do for j to nr do for k to grp do #F1:=evalm(F1+x^i*y^j*z*k*transmat(QV[i,j][k]));
```

```
> TT[i,j]:=TT[i,j]+1/det(J-t*transmat(QV[i,j][k]));

> od;od;od;

> print(F1);for i to np do for j to nr do print(i,j,simplify(factor(TT[i,j])),` = `,convert(TT[i,j],parfrac,t)) od;od;
```

$$\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$1,\ 1,\ \frac{4\,(t^3+t^2-t+1)}{(t-1)^4\,(t+1)^2\,(t^2+1)},\ \ =,\ \frac{1}{4\,(t+1)^2}+\frac{1}{t^2+1}+\frac{1}{(t-1)^4}-\frac{3}{4\,(t-1)}+\frac{3}{4\,(t+1)}+\frac{1}{4\,(t-1)^2}$$

$$1,\ 2,\ \frac{4\,(t^3+t^2-t+1)}{(t-1)^4\,(t+1)^2\,(t^2+1)},\ \ =,\ \frac{1}{4\,(t+1)^2}+\frac{1}{t^2+1}+\frac{1}{(t-1)^4}-\frac{3}{4\,(t-1)}+\frac{3}{4\,(t+1)}+\frac{1}{4\,(t-1)^2}$$

$$1,\ 3,\ \frac{4\,(t^3+t^2-t+1)}{(t-1)^4\,(t+1)^2\,(t^2+1)},\ \ =,\ \frac{1}{4\,(t+1)^2}+\frac{1}{t^2+1}+\frac{1}{(t-1)^4}-\frac{3}{4\,(t-1)}+\frac{3}{4\,(t+1)}+\frac{1}{4\,(t-1)^2}$$

```
> "group has",grp,"elements";evalm(QQ[1]);

> outfile:="/home/ph/maple/RCPermGroup.txml";
  writeto(outfile);print();

> printf(`<?xml version="1.0"?>`);
  print(` <!DOCTYPE html SYSTEM "mathml.dtd">`);
  print(` <html xmlns="http://www.w3.org/1999/xhtml"`);
  print(` xmlns:math="http://www.w3.org/1998/Math/MathML"`);
  print(` xmlns:xlink="http://www.w3.org/1999/xlink">`);
        print(`<body>`);

> appendto(outfile):

> unassign('vxv','QQQ','bb'):for i from 1 to grp do;

> print(cat(` g`,i,` :=`));print(`  `);

> print(convert(map(convert,matrans(QQ[i]),decimal,hex),'disjcyc'));print(`<p/>`);

>             vxv[i]:=convert(Matrix(QQ[i]),vector);
        od: writeto(terminal):interface(echo=1):

> bb:=Basis([seq(Vector(vxv[i]),i=1..grp)]):"linear dimension",nops(bb);

> QQQ:=seq(matrix(rk,rk,convert(bb[i],vector)),i =1..nops(bb)):
```

"group has", 4, "elements"

$$\begin{bmatrix} 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

outfile := "/home/ph/maple/RCPermGroup.txml"

"linear dimension", 4

```
> unassign('v','w','FM'):V:=matrix(rk,rk,(i,j)->v[i,j]):W:=matrix(rk,rk,(i,j)->w[i,j]):WW:=evalm(add(symult(QQ[k

> VV:=evalm(add(symmult(QQ[k],V),k=1..grp)):"Symmetric?",iszero(VV-transpose(VV));qvv:=checkspan([QQQ],
  in Alg(G)?",qvv[1]);

> EMM:=[evalm(map(diff,evalm(VV),[v[1,1]]))]:FMM:=[evalm(map(diff,evalm(WW),[w[1,1]]))]:

> for i to rk do for j to rk do FM:=map(diff,evalm(WW),[w[i,j]]);

> currk:=goRank(seq(EMM[k],k=1..nops(EMM))):EM:=map(diff,evalm(VV),[v[i,j]]);newrk:=goRank(EM,seq(EM

> if(newrk>currk) then EMM:=[op(EMM),evalm(EM)] fi;

> wcurrk:=goRank(seq(FMM[k],k=1..nops(FMM))):FM:=map(diff,evalm(WW),[w[i,j]]);wnewrk:=goRank(FM,seq

> if(wnewrk>wcurrk) then FMM:=[op(FMM),evalm(FM)]  fi;

> od;od;

> h:=vector(nops(EMM)):HZH:=evalm(add(h[k]*EMM[k],k=1..nops(EMM)));"Is Z in
  Vec(K)?",checkspan([QQQ],%);

> for i to nops(EMM) do m:=max(seq(EMM[i][1,kx],kx=1..rk)):print(i,"coeff",m):EMM[i]:=evalm(1/m*EMM[i])
  od:

> print("Basis for Z(G)");print(EMM);#G2:=evalm(add(sympow(QQ[k],2),k=1..grp)):
```

"Symmetric?", *false*

"Z(G) in Alg(G)?", *true*

$$HZH := \begin{bmatrix} h_1 & h_2 & h_3 & h_4 \\ h_2 & h_1 & h_4 & h_3 \\ h_4 & h_3 & h_1 & h_2 \\ h_3 & h_4 & h_2 & h_1 \end{bmatrix}$$

"Is Z in Vec(K)?", *true*, $[h_2, h_1, h_3, h_4]$

1, "coeff", 1

2, "coeff", 1

3, "coeff", 1

4, "coeff", 1

"Basis for Z(G)"

$$\left[\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&0&0&1\end{bmatrix},\begin{bmatrix}0&1&0&0\\1&0&0&0\\0&0&0&1\\0&0&1&0\end{bmatrix},\begin{bmatrix}0&0&1&0\\0&0&0&1\\0&1&0&0\\1&0&0&0\end{bmatrix},\begin{bmatrix}0&0&0&1\\0&0&1&0\\1&0&0&0\\0&1&0&0\end{bmatrix}\right]$$

```
> print(FMM);
```

$$\left[\begin{bmatrix}1&0&0&0\\0&1&0&0\\0&0&1&0\\0&0&0&1\end{bmatrix},\begin{bmatrix}0&1&0&0\\1&0&0&0\\0&0&0&1\\0&0&1&0\end{bmatrix},\begin{bmatrix}0&0&1&0\\0&0&0&1\\0&1&0&0\\1&0&0&0\end{bmatrix},\begin{bmatrix}0&0&0&1\\0&0&1&0\\1&0&0&0\\0&1&0&0\end{bmatrix}\right]$$

```
> for i to nops(EMM)-1 do for j from i+1 to nops(EMM) do print(i,j,iszero(cmm(EMM[i],EMM[j]))) od;od;
```

1, 2, *true*

1, 3, *true*

1, 4, *true*

2, 3, *true*

2, 4, *true*

3, 4, *true*

```
> eigs:=[]:for i to nops(EMM) do eigs:=[op(eigs),[eigenvalues(EMM[i])]]
  od:EIGS:=stackmatrix(seq(eigs[k],k=1..nops(EMM)));
```

$$EIGS := \begin{bmatrix}1&1&1&1\\1&-1&1&-1\\-1&1&I&-I\\-1&1&I&-I\end{bmatrix}$$

```
> TRACEMAT:=[seq(1,i=1..grp)]:
```

```
> for j from 1 to rk do  TRACEMAT:=stackmatrix(TRACEMAT,[seq(trace(sympow(QQ[i],j)),i=1..grp)])
  od:"Permutation chars",evalm(TRACEMAT);
```

```
> appendto(outfile):print(`<p/> <br/>PermChars:=<p/><center>`);
```

```
> goMatSimple(TRACEMAT);#for i to rk do print(row(TRACEMAT,i));
```

```
> print(`</center>`):print(` </body></html>`);
```

```
> writeto(terminal):interface(echo=1):
```

$$\text{"Permutation chars"}, \begin{bmatrix}1&1&1&1\\0&4&0&0\\2&6&0&0\\0&4&0&0\\1&1&1&1\end{bmatrix}$$

```
> unassign('EX'):Z1:=matrix(rk,n-rk,0):Z2:=matrix(n-rk,n,0):for i to nops(EMM) do
```

```
EX[i]:=stackmatrix(concat(EMM[i],Z1),Z2) od:

> unassign('XQX'):for i to np do for j to nr do XQX[i,j]:=[]:

> for k to nops(EMM) do XQX[i,j]:=[op(XQX[i,j]),multiply(C||i||j,EX[k],inverse(C||i||j))] od;
  od;od;evalm(add(XQX[1,1][k],k=1..nops(XQX[1,1])));
```

$$\begin{bmatrix} 1 & 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 & 1 \end{bmatrix}$$

```
> seq({i,choose(n,rk)[i]},i=1..binomial(n,rk));uu:=vector(binomial(n,rk),1):

{1, [1, 2, 3, 4]}, {2, [1, 2, 3, 5]}, {3, [1, 2, 3, 6]}, {4, [1, 2, 4, 5]}, {5, [1, 2, 4, 6]}, {6, [1, 2, 5, 6]}, {7, [1, 3, 4, 5]},
  {8, [1, 3, 4, 6]}, {9, [1, 3, 5, 6]}, {10, [1, 4, 5, 6]}, {11, [2, 3, 4, 5]}, {12, [2, 3, 4, 6]}, {13, [2, 3, 5, 6]}, {14, [2, 4, 5, 6
  {15, [3, 4, 5, 6]}

> for i to np do "PT"||i=PT||i od;for i to nr do "RG"||i=RG||i od;

"PT1" = {{1}, {2, 3}, {5, 6}, {4}}

"RG1" = {1, 3, 4, 6}

"RG2" = {1, 3, 4, 5}

"RG3" = {1, 2, 4, 5}

> ############ Range Mats

> unassign('RGMAT'):for j to nr do map(simplify,multiply(rk*pi,E1||j));RGMAT||j:=diag(seq(%[k],k=1..n)) od:

> ############  Zeon powers

> unassign('G2'):for j to np do for k to nr do for l to rk do G2[j,k,l]:=matrix(n,n,0);od:od: od;

> for i to np do for j to nr do for kr to rk do

> G2[i,j,kr]:=evalm(add(sympow(transmat(QV[i,j][k]),kr),k=1..grp)/grp):#print(i,j,%);

> od;od;od;
```

U series

```
> unassign('Y'):for kr from rk-1 to 1 by -1 do print("LEVEL",kr);EP(kr,kr+1):

> for i to np do for j to nr do Y[i,j,kr]:=sympow(E||i||j,kr): od: od: od:

"LEVEL", 3

"LEVEL", 2

"LEVEL", 1

> for i to np do for j to nr do Y[i,j,rk]:=sympow(E||i||j,rk) ;od;od;
```

> for i to np do for j to nr do

> for kr from rk-1 to 1 by -1 do print("LEVEL",kr):

> nn:=binomial(n,kr):uuu:=vector(binomial(n,kr+1),1):uue:=vector(nn,1);

> Z:=multiply(PEI||kr||(kr+1),Y[i,j,kr+1]);v1:=multiply(Z,uuu):v2:=multiply(Y[i,j,kr],uue):print(i,j,v1,v2);print

"LEVEL", 3

$1, 1, \left[0, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}, 0, 0, 0, 0, \frac{1}{4}, \frac{1}{4}, 0, \frac{1}{4}, \frac{1}{4}, 0, 0\right], [0, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 1, 1, 0, 1, 1, 0, 0]$

{2, 3, 4, 5, 6, 7, 8, 9, 14, 15, 17, 18}

{2, 3, 4, 5, 6, 7, 8, 9, 14, 15, 17, 18}

"LEVEL", 2

$1, 1, \left[\frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, 0, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, 0\right], [1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 0]$

{1, 2, 3, 4, 5, 7, 8, 9, 10, 11, 12, 13, 14}

{1, 2, 3, 4, 5, 7, 8, 9, 10, 11, 12, 13, 14}

"LEVEL", 1

$1, 1, \left[\frac{3}{4}, \frac{3}{4}, \frac{3}{4}, \frac{3}{4}, \frac{3}{4}, \frac{3}{4}\right], [1, 1, 1, 1, 1, 1]$

{1, 2, 3, 4, 5, 6}

{1, 2, 3, 4, 5, 6}

"LEVEL", 3

$1, 2, \left[0, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}, 0, 0, 0, 0, \frac{1}{4}, \frac{1}{4}, 0, \frac{1}{4}, \frac{1}{4}, 0, 0\right], [0, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 1, 1, 0, 1, 1, 0, 0]$

{2, 3, 4, 5, 6, 7, 8, 9, 14, 15, 17, 18}

{2, 3, 4, 5, 6, 7, 8, 9, 14, 15, 17, 18}

"LEVEL", 2

$1, 2, \left[\frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, 0, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, 0\right], [1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 0]$

{1, 2, 3, 4, 5, 7, 8, 9, 10, 11, 12, 13, 14}

{1, 2, 3, 4, 5, 7, 8, 9, 10, 11, 12, 13, 14}

"LEVEL", 1

$1, 2, \left[\frac{3}{4}, \frac{3}{4}, \frac{3}{4}, \frac{3}{4}, \frac{3}{4}, \frac{3}{4}\right], [1, 1, 1, 1, 1, 1]$

{1, 2, 3, 4, 5, 6}

{1, 2, 3, 4, 5, 6}

"LEVEL", 3

$1, 3, \left[0, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}, 0, 0, 0, 0, \frac{1}{4}, \frac{1}{4}, 0, \frac{1}{4}, \frac{1}{4}, 0, 0\right], [0, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 1, 1, 0, 1, 1, 0, 0]$

{2, 3, 4, 5, 6, 7, 8, 9, 14, 15, 17, 18}

{2, 3, 4, 5, 6, 7, 8, 9, 14, 15, 17, 18}

"LEVEL", 2

$1, 3, \left[\frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, 0, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, 0\right], [1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 0]$

{1, 2, 3, 4, 5, 7, 8, 9, 10, 11, 12, 13, 14}

{1, 2, 3, 4, 5, 7, 8, 9, 10, 11, 12, 13, 14}

"LEVEL", 1

$1, 3, \left[\frac{3}{4}, \frac{3}{4}, \frac{3}{4}, \frac{3}{4}, \frac{3}{4}, \frac{3}{4}\right], [1, 1, 1, 1, 1, 1]$

{1, 2, 3, 4, 5, 6}

{1, 2, 3, 4, 5, 6}

Pi Series

```
> for kr from rk to 1 by -1 do print("LEVEL",kr);
> for i to np do for j to nr do
> X:=row(multiply(sympow(UJ,kr)/(kr)!,G2[i,j,kr]),1);print(i,j);print(X);
> od;od;"NEXT LEVEL?",map(simplify,piV(X,kr)) od;
```

"LEVEL", 4

1, 1

[0, 0, 0, 0, 0, 0, 0, 4, 0, 0, 0, 0, 0, 0, 0]

1, 2

[0, 0, 0, 0, 0, 0, 4, 0, 0, 0, 0, 0, 0, 0, 0]

1, 3

[0, 0, 0, 4, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0]

"NEXT LEVEL?", [0, 4, 4, 0, 0, 0, 0, 4, 0, 0, 0, 0, 0, 4, 0, 0, 0, 0, 0, 0]

"LEVEL", 3

1, 1

[0, 0, 0, 0, 3, 0, 3, 0, 3, 0, 0, 0, 0, 0, 0, 0, 0, 3, 0, 0]

1, 2

[0, 0, 0, 0, 3, 3, 0, 3, 0, 0, 0, 0, 0, 0, 0, 0, 3, 0, 0, 0]

1, 3

[0, 3, 3, 0, 0, 0, 0, 3, 0, 0, 0, 0, 0, 3, 0, 0, 0, 0, 0, 0]

"NEXT LEVEL?", [6, 0, 6, 6, 0, 0, 6, 6, 0, 0, 0, 0, 6, 0, 0]

"LEVEL", 2

1, 1

$$\left[0, 2, \frac{9}{4}, 0, \frac{9}{4}, 0, 0, 0, 0, \frac{9}{4}, 0, \frac{9}{4}, 0, 2, 0\right]$$

1, 2

$$\left[0, 2, \frac{9}{4}, \frac{9}{4}, 0, 0, 0, 0, 0, \frac{9}{4}, \frac{9}{4}, 0, 2, 0, 0\right]$$

1, 3

$$\left[2, 0, \frac{9}{4}, \frac{9}{4}, 0, 0, \frac{9}{4}, \frac{9}{4}, 0, 0, 0, 0, 2, 0, 0\right]$$

$$\text{"NEXT LEVEL?"}, \left[\frac{13}{2}, \frac{13}{2}, 0, \frac{13}{2}, \frac{13}{2}, 0\right]$$

"LEVEL", 1

1, 1

$$\left[\frac{3}{2}, 0, \frac{3}{2}, \frac{3}{2}, 0, \frac{3}{2}\right]$$

1, 2

$$\left[\frac{3}{2}, 0, \frac{3}{2}, \frac{3}{2}, \frac{3}{2}, 0\right]$$

1, 3

$$\left[\frac{3}{2}, \frac{3}{2}, 0, \frac{3}{2}, \frac{3}{2}, 0\right]$$

"NEXT LEVEL?", [6]

> "pi2",clearDenoms(pi2);"u2",clearDenoms(u2);

> print("pi",pi);

"pi2", [8, 24, 36, 18, 18, 0, 9, 9, 0, 27, 9, 18, 16, 16, 0]

"u2", [1, 1, 1, 1, 1, 0, 1, 1, 1, 1, 1, 1, 1, 1, 0]

$$\text{"pi"}, \left[\frac{1}{4}, \frac{1}{16}, \frac{3}{16}, \frac{1}{4}, \frac{1}{8}, \frac{1}{8}\right]$$

> unassign('piv','uv','RkR','BkB'):

> #AK:=evalm((1/2)*(RK+BK)):

```
> piv||rk:=clearDenoms(evalm(1/x[1]*(linsolve(transpose(JK-AK),vector(NK,0),'rr',x)))):#readVec(%);
```

```
> uv||rk:=clearDenoms(evalm((1/x[1]*linsolve(JK-AK,vector(NK,0),'rr',x)))):#readVec(%);
```

```
> unassign('ax','cx'):for m to NK do ax[m]:=vector(n,0):for j to n do if(evalb(j in choose(n,rk)[m]))  then
  ax[m][j]:=1 fi od: od:
```

```
> picheck:=evalm(add((piv||rk)[cx]*ax[cx],cx=1..NK)):"picheck",iszero(clearDenoms(picheck)-clearDenoms(pi))
```

"picheck", *true*

```
> seq({i,choose(n,2)[i]},i=1..binomial(n,rk));uu:=vector(binomial(n,2),1):
```

{1, [1, 2]}, {2, [1, 3]}, {3, [1, 4]}, {4, [1, 5]}, {5, [1, 6]}, {6, [2, 3]}, {7, [2, 4]}, {8, [2, 5]}, {9, [2, 6]}, {10, [3, 4]},
{11, [3, 5]}, {12, [3, 6]}, {13, [4, 5]}, {14, [4, 6]}, {15, [5, 6]}

```
> print("pi"||rk,piv||rk);print("u"||rk,uv||rk);
```

```
> for i from rk-1 to 1 by -1 do
  piv||i:=map(simplify,piV(piv||(i+1),i+1));uv||i:=map(simplify,uV(uv||(i+1),i+1,pi));
```

```
> print("pi"||i,piv||i);print("u"||i,uv||i);
```

```
> RkR:=sympow(R,i);BkB:=sympow(B,i);JKJ:=sympow(J,i):AKA:=evalm((1/2)*(RkR+BkB)):
```

```
> print(iszero(multiply(JKJ-AKA,uv||i)));print(iszero(multiply(piv||i,JKJ-AKA)));od:
```

"pi4", [0, 0, 0, 1, 0, 0, 1, 2, 0, 0, 0, 0, 0, 0, 0]

"u4", [0, 0, 0, 1, 1, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0]

"pi3", [0, 1, 1, 0, 3, 1, 2, 2, 2, 0, 0, 0, 0, 1, 0, 0, 1, 2, 0, 0]

"u3", $\left[0, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}, 0, 0, 0, 0, \frac{1}{4}, \frac{1}{4}, 0, \frac{1}{4}, \frac{1}{4}, 0, 0\right]$

true

true

"pi2", [2, 6, 8, 4, 4, 0, 2, 2, 0, 6, 2, 4, 4, 4, 0]

"u2", $\left[\frac{1}{8}, \frac{1}{8}, \frac{1}{8}, \frac{1}{8}, \frac{1}{8}, 0, \frac{1}{8}, \frac{1}{8}, \frac{1}{8}, \frac{1}{8}, \frac{1}{8}, \frac{1}{8}, \frac{1}{8}, \frac{1}{8}, 0\right]$

true

true

"pi1", [24, 6, 18, 24, 12, 12]

"u1", $\left[\frac{3}{32}, \frac{3}{32}, \frac{3}{32}, \frac{3}{32}, \frac{3}{32}, \frac{3}{32}\right]$

true

true

```
> for i from rk to 1 by -1 do
```

```
> RkR:=sympow(R,i);BkB:=sympow(B,i);JKJ:=sympow(J,i):AKA:=evalm((1/2)*(RkR+BkB)):NA:=binomial(n,i):
> piv||i:=clearDenoms(evalm(1/x[1]*(linsolve(transpose(JKJ-AKA),vector(NA,0),'rr',x))));
> print("pi"||i,piv||i);
> if(i<rk) then X:=map(simplify,piV(piv||(i+1),i+1));print("PI",X);print(iszero(multiply(X,JKJ-AKA)));fi;
> uv||i:=clearDenoms(evalm((1/x[1]*linsolve(JKJ-AKA,vector(NA,0),'rr',x))));
> print("u"||i,uv||i);
> if(i<rk) then Y:=map(simplify,uV(uv||(i+1),i+1,pi));
> print("U",Y);print(iszero(multiply(JKJ-AKA,Y)));fi; od:
```

"pi4", [0, 0, 0, 1, 0, 0, 1, 2, 0, 0, 0, 0, 0, 0, 0]

"u4", [0, 0, 0, 1, 1, 0, 1, 1, 0, 0, 0, 0, 0, 0, 0]

"pi3", [0, 1, 1, 0, 3, 1, 2, 2, 2, 0, 0, 0, 0, 1, 0, 0, 1, 2, 0, 0]

"PI", [0, 1, 1, 0, 3, 1, 2, 2, 2, 0, 0, 0, 0, 1, 0, 0, 1, 2, 0, 0]

true

"u3", [0, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 1, 1, 0, 1, 1, 0, 0]

$$\text{"U"}, \left[0, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}, \frac{1}{4}, 0, 0, 0, 0, \frac{1}{4}, \frac{1}{4}, 0, \frac{1}{4}, \frac{1}{4}, 0, 0\right]$$

true

$$\text{"pi2"}, [x_2, 3\,x_2, 4\,x_1, 2\,x_1, 2\,x_1, 0, x_1, x_1, 0, 3\,x_1, x_1, 2\,x_1, 2\,x_2, 2\,x_2, 0]$$

"PI", [2, 6, 8, 4, 4, 0, 2, 2, 0, 6, 2, 4, 4, 4, 0]

true

$$\text{"u2"}, [x_2, x_2, x_1, x_1, x_1, 0, x_1, x_1, x_1, x_1, x_1, x_1, x_2, x_2, 0]$$

$$\text{"U"}, \left[\frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, 0, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{2}, 0\right]$$

true

"pi1", [4, 1, 3, 4, 2, 2]

$$\text{"PI"}, [4\,x_2 + 8\,x_1, x_2 + 2\,x_1, 3\,x_2 + 6\,x_1, 4\,x_2 + 8\,x_1, 4\,x_1 + 2\,x_2, 4\,x_1 + 2\,x_2]$$

true

"u1", [1, 1, 1, 1, 1, 1]

$$\text{"U"}, \left[\frac{1}{4}\,x_2 + \frac{1}{2}\,x_1, \frac{1}{4}\,x_2 + \frac{1}{2}\,x_1, \frac{1}{4}\,x_2 + \frac{1}{2}\,x_1, \frac{1}{4}\,x_2 + \frac{1}{2}\,x_1, \frac{1}{4}\,x_2 + \frac{1}{2}\,x_1, \frac{1}{4}\,x_2 + \frac{1}{2}\,x_1\right]$$

true

```
> for kr to rk do
> G1[kr]:=evalm(add(sympow(QQ[k],kr),k=1..grp)/grp):
> od;
```

$$G1_1 := \begin{bmatrix} \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \end{bmatrix}$$

$$G1_2 := \begin{bmatrix} \frac{1}{2} & 0 & 0 & 0 & 0 & \frac{1}{2} \\ 0 & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & 0 \\ 0 & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & 0 \\ 0 & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & 0 \\ 0 & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & 0 \\ \frac{1}{2} & 0 & 0 & 0 & 0 & \frac{1}{2} \end{bmatrix}$$

$$G1_3 := \begin{bmatrix} \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \\ \frac{1}{4} & \frac{1}{4} & \frac{1}{4} & \frac{1}{4} \end{bmatrix}$$

$$G1_4 := [1]$$

```
> print(TRACEMAT);
```

$$\begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & 4 & 0 & 0 \\ 2 & 6 & 0 & 0 \\ 0 & 4 & 0 & 0 \\ 1 & 1 & 1 & 1 \end{bmatrix}$$

```
>
```