```
? PermutationProduct
```

PermutationProduct[*a*, *b*, *c*] gives the product of permutations *a*, *b*, *c*. ≫

```
PP[a_List] := a;
(*PP[a_List, b_List] := Table[b[[j]],{j,a}];*)
PP[a_List, b_List] := b[[a]];
(*PP[a_List,b_List, rest__List] := PP[PP[a,b],rest];*)
PP[a_List, rest__List] := PP[a, PP@rest];
```

```
PP[{2, 3, 1}, {2, 1, 3}]
```

```
{1, 3, 2}
```

```
PP[{2, 3, 1}, {2, 1, 3}, {3, 2, 1}]
```

```
{3, 1, 2}
```

```
{ian, huan, justin}[[2]]
```

```
huan
```

```
{ian, huan, justin}[[{2, 3, 1}]]
```

```
{huan, justin, ian}
```

```
? InversePermutation
```

InversePermutation[*perm*] returns the inverse of permutation *perm*. ≫

```
Position[{huan, {justin, adam}, ian}, justin]
```

```
{{2, 1}}
```

```
IP[a_List] := Table[
  Position[a, i][[1, 1]],
  {i, Length@a}
 ]
```

```
IP[{2, 3, 1}]
```

```
{3, 1, 2}
```

```
0 {3, 1, 2}
```

```
{0, 0, 0}
```

```
IP[a_List] := (
  t = 0 a;
  Do[t[[a[[i]]]] = i, {i, Length@a}];
  t
 )
```

```
IP[{2, 3, 1}]
```

```
{3, 1, 2}
```

```
(*IP[a_List] := Ordering[a]*)
```

```
? PermutationSupport
```

PermutationSupport[*perm*] returns the support of the permutation *perm*. ≫

```
PS[a_] := Join @@ Position[a - Range[Length@a], Except[0], {1}, Heads → False]
```

```
PS[{1, 2, 4, 3, 5}]
```

```
{3, 4}
```

```
PS[a_] := Join @@ Position[a - Range[Length@a], i_Integer /; i ≠ 0]
```

```
PS[{1, 2, 4, 3, 5}]
```

```
{3, 4}
```

```
PS[a_] := Select[a, # ≠ a[[#]] &]
```

```
PS[{1, 2, 4, 3, 5}]
```

```
{4, 3}
```

```
Trace[IP[{2, 3, 1}]] // Column
```

```
IP[{2, 3, 1}]
t = 0 {2, 3, 1}; Do[t[[{2, 3, 1}[[i]]]] = i, {i, Length[{2, 3, 1}]}]; t
{{0 {2, 3, 1}, {0 × 2, 0 × 3, 0 × 1}, {0 × 2, 0}, {0 × 3, 0}, {0 × 1, 0}, {0, 0, 0}},
 t = {0, 0, 0}, {0, 0, 0}}
{Do[t[[{2, 3, 1}[[i]]]] = i, {i, Length[{2, 3, 1}]}],
 {Length[{2, 3, 1}], 3}, {{i, 1}, t[[{2, 3, 1}[[i]]]] = 1, 1},
 {{i, 2}, t[[{2, 3, 1}[[i]]]] = 2, 2}, {{i, 3}, t[[{2, 3, 1}[[i]]]] = 3, 3}, Null}
{t, {3, 1, 2}}
{3, 1, 2}
```

```
? PermutationReplace
```

PermutationReplace[*expr*, *perm*] replaces each part in *expr* by its image under the permutation *perm*.

PermutationReplace[*expr*, *gr*] returns the list of images of *expr* under all elements of the permutation group *gr*. ≫

```
PermutationReplace[#, Cycles[{{1, 2, 3}}]] & /@ {
  1, lennart, lennart[1, 3], {3, 4}, lennart[{1, 2}, 3]
 }
```

```
{2, lennart, lennart[2, 1], {1, 4}, lennart[{2, 3}, 1]}
```

```
PR[expr_Integer, a_List] := If[1 ≤ expr ≤ Length[a], a[[expr]], expr];
PR[head_[elements___], a_List] := PR[head, a] @@ (PR[#, a] & /@ {elements});
PR[expr_, a_List] := expr;
```

```
PR[#, {2, 3, 1}] & /@ {
  1, lennart, lennart[1, 3], {3, 4}, lennart[{1, 2}, 3]
 }
```

```
{2, lennart, lennart[2, 1], {1, 4}, lennart[{2, 3}, 1]}
```

```
Sequence[1, 2, 3]
```

```
Sequence[1, 2, 3]
```

```
just[1, 2, Sequence[3, 4]]
```

```
just[1, 2, 3, 4]
```

```
sq[x_] := x^2;
sq /@ {1, 2, 3}
```

```
{1, 4, 9}
```

```
sq /@ f[1, 2, 3]
```

```
f[1, 4, 9]
```

```
Map[sq, {3, 4, 5}]
```

```
{9, 16, 25}
```

```
sq /@ Sequence[1, 2, 3]
```

```
Map::nonopt : Options expected (instead of 3) beyond position 3 in Map[sq, 1, 2, 3]. An option must be a rule or a list of rules. >>
```

```
Map[sq, 1, 2, 3]
```

```
PR[expr_, a_List] := expr /. Table[i → a[[i]], {i, a}]
```

```
PR[#, {2, 3, 1}] & /@ {
  1, lennart, lennart[1, 3], {3, 4}, lennart[{1, 2}, 3]
 }
```

```
{2, lennart, lennart[2, 1], {1, 4}, lennart[{2, 3}, 1]}
```

```
a = {2, 3, 1};
Table[i → a[[i]], {i, Length@a}]
```

```
{1 → 2, 2 → 3, 3 → 1}
```

```
Range@Length@a → a
```

```
{1, 2, 3} → {2, 3, 1}
```

```
Transpose[{{1, 2, 3}, {4, 5, 6}}]
```

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

```
Transpose[Range@Length@a → a]
```

```
Transpose[{1, 2, 3} → {2, 3, 1}]
```

```
Transpose[{Range@Length@a , a}]
```

```
{{1, 2}, {2, 3}, {3, 1}}
```

```
Rule /@ Transpose[{Range@Length@a , a}]
```

```
Rule::argr : Rule called with 1 argument; 2 arguments are expected. >>

Rule::argr : Rule called with 1 argument; 2 arguments are expected. >>

Rule::argr : Rule called with 1 argument; 2 arguments are expected. >>

General::stop : Further output of Rule::argr will be suppressed during this calculation. >>

{Rule[{1, 2}], Rule[{2, 3}], Rule[{3, 1}]}
```

```
Rule @@@ Transpose[{Range@Length@a , a}]
```

```
{1 → 2, 2 → 3, 3 → 1}
```

```
Thread[Range@Length@a → a]
```

```
{1 → 2, 2 → 3, 3 → 1}
```

```
Thread[a → a[[a]]]
```

```
{2 → 3, 3 → 1, 1 → 2}
```

```
PR[expr_, a_List] := expr /. Thread[a → a[[a]]]
```

```
PR[#, {2, 3, 1}] & /@ {
  1, lennart, lennart[1, 3], {3, 4}, lennart[{1, 2}, 3]
 }
```

```
{2, lennart, lennart[2, 1], {1, 4}, lennart[{2, 3}, 1]}
```

```
n = 54;
g1 = Cycles[{{1, 18, 45, 28}, {2, 27, 44, 19},
     {3, 36, 43, 10}, {46, 52, 54, 48}, {47, 49, 53, 51}}];
g2 = Cycles[{{7, 16, 39, 30}, {8, 25, 38, 21}, {9, 34, 37, 12},
     {13, 15, 33, 31}, {14, 24, 32, 22}}];
g3 = Cycles[{{28, 31, 34, 48}, {29, 32, 35, 47}, {30, 33, 36, 46},
     {37, 39, 45, 43}, {38, 42, 44, 40}}];
g4 = Cycles[{{1, 3, 9, 7}, {2, 6, 8, 4}, {10, 54, 16, 13},
     {11, 53, 17, 14}, {12, 52, 18, 15}}];
g5 = Cycles[{{1, 13, 37, 46}, {4, 22, 40, 49}, {7, 31, 43, 52},
     {10, 12, 30, 28}, {11, 21, 29, 19}}];
g6 = Cycles[{{3, 48, 39, 15}, {6, 51, 42, 24}, {9, 54, 45, 33},
     {16, 18, 36, 34}, {17, 27, 35, 25}}];
```

```
σ_ ∘ τ_ := PP[τ, σ];
Feed[Range[n]] := 1 + 1;
Feed[τ_] := Module[{i, j, k, l},
   i = Min[PS[τ]];
   j = PR[i, τ];
   If[Head[σ_{i,j}] === List,
    Feed[IP[σ_{i,j}] ∘ τ],
    (*Else*) σ_{i,j} = τ;
    For[k = 1, k < n, ++k,
     For[l = k + 1, l ≤ n, ++l,
      If[Head[σ_{k,l}] === List,
       Feed[σ_{i,j} ∘ σ_{k,l}]; Feed[σ_{k,l} ∘ σ_{i,j}]]
     ]]
   ]];
$RecursionLimit = ∞;
```

```
Table[γ_α = PermutationList[g_α, n], {α, 6}] // MatrixForm
```

$$\begin{pmatrix} 18 & 27 & 36 & 4 & 5 & 6 & 7 & 8 & 9 & 3 & 11 & 12 & 13 & 14 & 15 & 16 & 17 & 45 & 2 & 20 & 21 & 22 & 23 & 24 & 25 \\ 1 & 2 & 3 & 4 & 5 & 6 & 16 & 25 & 34 & 10 & 11 & 9 & 15 & 24 & 33 & 39 & 17 & 18 & 19 & 20 & 8 & 14 & 23 & 32 & 38 \\ 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 15 & 16 & 17 & 18 & 19 & 20 & 21 & 22 & 23 & 24 & 25 \\ 3 & 6 & 9 & 2 & 5 & 8 & 1 & 4 & 7 & 54 & 53 & 52 & 10 & 11 & 12 & 13 & 14 & 15 & 19 & 20 & 21 & 22 & 23 & 24 & 25 \\ 13 & 2 & 3 & 22 & 5 & 6 & 31 & 8 & 9 & 12 & 21 & 30 & 37 & 14 & 15 & 16 & 17 & 18 & 11 & 20 & 29 & 40 & 23 & 24 & 25 \\ 1 & 2 & 48 & 4 & 5 & 51 & 7 & 8 & 54 & 10 & 11 & 12 & 13 & 14 & 3 & 18 & 27 & 36 & 19 & 20 & 21 & 22 & 23 & 6 & 17 \end{pmatrix}$$

```
Table[Feed[γ_α];
  ∏_{i=1}^{n} (1 + Count[Range[n], j_ /; Head[σ_{i,j}] == List]), {α, 6}] // Timing
```

```
{0.03125, {43 252 003 274 489 856 000, 43 252 003 274 489 856 000, 43 252 003 274 489 856 000,
  43 252 003 274 489 856 000, 43 252 003 274 489 856 000, 43 252 003 274 489 856 000}}
```

```
Feed[γ_1]
```

```
2
```