

```
<< KnotTheory`
Loading KnotTheory` version
of September 6, 2014, 13:37:37.2841.
Read more at http://katlas.org/wiki/KnotTheory.
```

**Loading KnotTheory`** **Table[K → V<sub>3</sub>[K], {K, AllKnots@{3, 7}}]** **Computing V<sub>3</sub>**

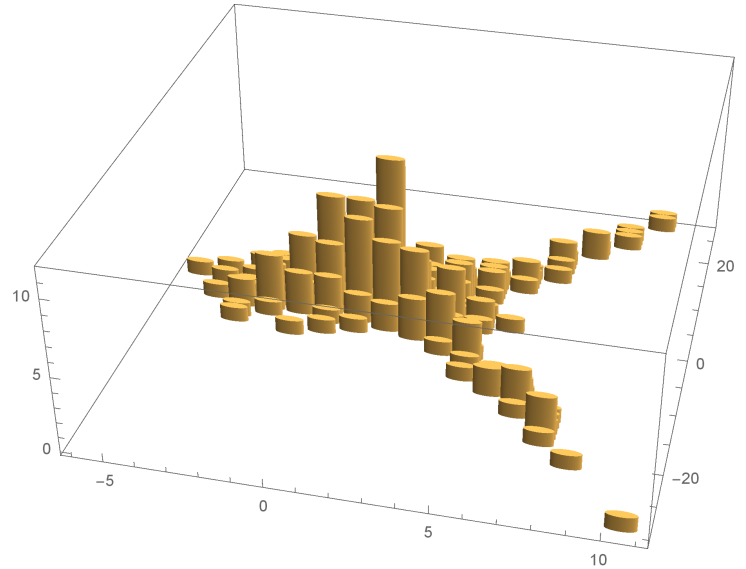
```
{31 → -1, 41 → 0, 51 → -5, 52 → -3, 61 → 1, 62 → 1, 63 → 0,
71 → -14, 72 → -6, 73 → 11, 74 → 8, 75 → -8, 76 → -2, 77 → -1}
```

**GD[g\_GD] := g;** **Gauss Diagram Utilities**

```
GD[L_] := GD@@PD[L] /.
X[i_, j_, k_, l_] => If[PositiveQ@X[i, j, k, l],
Api,i, Amj,i];
Draw[g_GD] := Module[{n = Max@Cases[g, _Integer, ∞]},
Graphics[{
Line[{{0, 0}, {n+1, 0}}],
List@@g /. (ah_)i,j => {
Arrow[BezierCurve[{{i, 0}, {i+j, Abs[j-i]}/2,
{j, 0}}]],
Text[ah /. {Ap → "+", Am → "-"}, {i, 0.3}]},
Table[Text[i, {i, -0.5}], {i, n}]]]
```

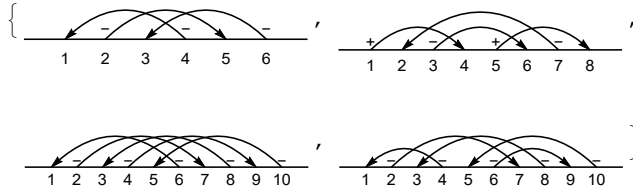
**Histogram3D[** **Willerton's Fish**

```
Table[{V2[K], V3[K]}, {K, AllKnots@{3, 10}}],
{1}]
```



**Draw /@ GD /@ AllKnots@{3, 5}** **Some Gauss Diagrams**

KnotTheory::loading: Loading precomputed data in PD4Knots`.



**GD /@ AllKnots@{3, 5}** **Some Gauss Diagrams, 2**

```
{GD[Am4,1, Am6,3, Am2,5], GD[Ap1,4, Ap5,8, Am3,6, Am7,2],
GD[Am6,1, Am8,3, Am10,5, Am2,7, Am4,9],
GD[Am4,1, Am8,3, Am10,5, Am6,9, Am2,7}
```

**CF[g\_GD] := Sort[** **V<sub>2</sub> Definition**

```
g /. Thread[Sort@Cases[g, _Integer, ∞] →
Range[2 Length[g]]];
PV[F_GD, g_GD] /; Length[F] > Length[g] := 0;
PV[F_GD, g_GD] /; Length[F] < Length[g] := Sum[
PV[F, y], {y, Subsets[g, {Length[F]}]}];
PV[F_GD, g_GD] /; Length[F] == Length[g] := If[
CF[F] == CF[g /. Ap | Am → A], (-1)Count[g, Am_], 0];
V2[g_] := V2[g] = PV[GD[A3,1, A2,4], GD[g]];
```

**Format[Knot[n\_, k\_]] := nk;** **Computing V<sub>2</sub>**

```
Table[K → V2[K], {K, AllKnots@{3, 7}}]
{31 → 1, 41 → -1, 51 → 3, 52 → 2, 61 → -2, 62 → -1, 63 → 1,
71 → 6, 72 → 3, 73 → 5, 74 → 4, 75 → 4, 76 → 1, 77 → -1}
```

**PV[F1\_ + F2\_, g\_] := PV[F1, g] + PV[F2, g];** **V<sub>3</sub> Definition**

```
PV[c_*F_GD, g_] := c PV[F, g];
ρk[g_] := g /. _Integer → Mod[i - k, 2 Length@g, 1];
F3 = ∑k=05 (3 ρk@GD[A1,5, A4,2, A6,3] + 2 ρk@GD[A1,4, A5,2, A3,6]);
V3[K_] := V3[K] = PV[F3, GD@K] / 6;
```

**G[λ]<sub>a,b</sub> := ∂<sub>t<sub>a</sub>, h<sub>b</sub></sub> λ;** **Gassner Utilities**

```
G /: Factor[G[λ_]] :=
G[Collect[λ, h_, Collect[#, t_, Factor] &]];
Format@γ_G := Module[{S = Union@Cases[γ, (h | t)a → a, ∞]},
Table[γa,b, {a, S}, {b, S}] // MatrixForm];
```

**G /: G[λ1\_] G[λ2\_] := G[λ1 + λ2];** **The Gassner Program**

```
ma,b → c[G[λ_]] := Module[{α, β, γ, δ, θ, ε, φ, ψ, Ξ, μ},
{
α β θ
γ δ ε
φ ψ Ξ
} = {
∂ta, ha λ ∂ta, hb λ ∂ta λ
∂tb, ha λ ∂tb, hb λ ∂tb λ
∂ha λ ∂hb λ λ
} /. (t | h)a|b → 0;
μ = 1 - β;
G[Tr[{tc
1} T. (γ + α δ / μ ε + δ θ / μ). (hc
1)]] /. Ta|b → Tc //
Factor];
Rpa,b := G[Tr[{ta
tb} T. (1 1 - Ta
0 Ta) . (ha
hb)]];
Rma,b := Rpa,b /. Ta → 1 / Ta;
```

**GG[g\_GD, k\_, F\_, BB\_] :=** **The Gauss-Gassner-Program**

```
Module[{n = 2 Length@g + Length@BB, y, cuts, rr, γ0, γ},
γ0 = G[tn+1 hn+1] Times @@ g /. {Ap → Rp, Am → Rm};
γ0 *= G[Sum[βa,b ta hb, {a, BB}, {b, BB}]];
Sum[γ = γ0;
cuts = Cases[y, _Integer, ∞] ∪ {n+1};
rr = Thread[cuts → Range[Length@cuts]];
Do[If[! MemberQ[cuts, j], γ = γ / mj, j+1+j+1], {j, n}];
F[y /. rr, γ /. (vn)a → va/rr,
(*over*) {y, Subsets[List@g, k]}]];
GG[g_GD, k_, F_] := GG[g, k, F, {}];
```