

1. [Gamalt prófdæmi] Hér fyrir neðan er tvívíða þríhyrningnum með hornpunkta $(0,0)$, $(1,0)$ og $(0,1)$ varpað yfir í þríhyrninginn með hornpunkta $(4,4)$, $(4,2)$ og $(2,4)$. Athugið að röð hornpunktanna skiptir máli. Sýnið samsetta vörpun sem framkvæmir þessa breytingu. Ekki þarf að sýna fylkin sjálf, en rökstyðjið hvern lið í samsettu vörpuninni.

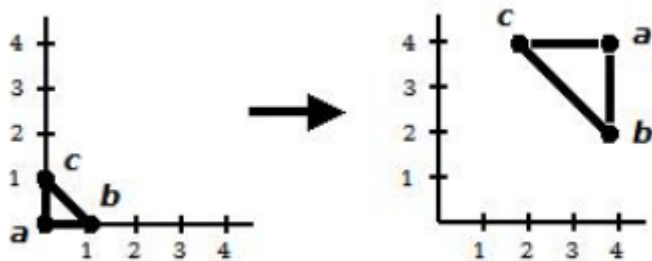


Figure 1: Vörpun

- Byrjum á því að stilla stærðina og speglum á y-ásinn til að skipta um röðun fyrir C og B. $S(2, -2)$
- Snúum svo til að færa hornpunktana yfir á réttan form. $R(270)$
- Kvörðum síðast til að færa hnutana yfir á réttan stað. $T(4, 4)$
- **Vörpun er þá** $M = T(4, 4)R(270)S(2, -2)$

2. [Byggt á gömlu prófdæmi] Gefinn er þrívíður teningur með gagnstæð horn í $(-4, -1, 2)$ og $(-1, 2, -1)$. Það á að breyta honum í einingartening með miðju á sama stað og upphaflegi teningurinn og honum hefur verið snúið um θ gráður rangsælis um y ás. Sýnið samsettu vörpununina sem útfærir þessa breytingu á teningnum og útskýrið einstakar varpanir í henni. Það þarf ekki að gefa upp fylkin sjálf, heldur bara nefna hverja vörpun og gildin í henni.

- Viljum halda miðju á sama stað, skilgreinum miðjupúnt $p = [-2.5, 0.5, 0.5]$
- Getum þá hliðrað niður í núllpunkt $T(-p)$
- Breytum stærðina til að fá einingartening $S(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})$
- Framkvæmum rangsælis snúningu $R_y(\theta)$
- Hliðrum til baka $T(p)$
- **Erum þá með** $M = T(p)R_y(\theta)S(\frac{1}{3}, \frac{1}{3}, \frac{1}{3})T(-p)$

3. Skrifid WebGL forrit sem býr til tölvuskjá. Í okkar útgáfu af tölvuskjá er hann samsettur úr þremur teningum. Það er skjáinn sjálfur, sem er flatur og breiður “teningur”, síðan er standurinn samsettur úr tveimur teningum. Annar þeirra er nokkur flatur og liggur á jörðinni, en hinn er eins og þykk stöng sem tengir botninn við skjáinn sjálfan. Annars treysti ég því að þið vitið hvernig tölvuskjár lítur út! Notandinn á að geta snúið skjánum í hringi með músinni (svipað og í [letterH](#)). Skilið skjámynd og hlekk á forritið ykkar.

[Hlekkur](#)

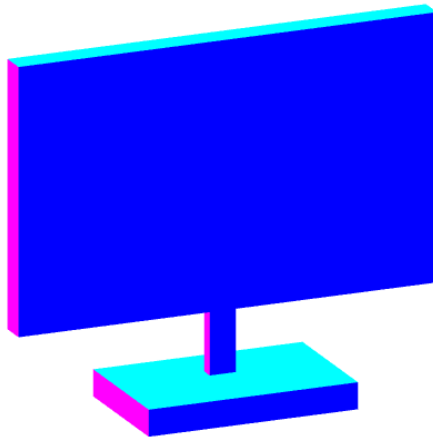


Figure 2: Skjáskot: Tölvuskjá

```
var mv = mat4();
mv = mult(mv, rotateX(spinX));
mv = mult(mv, rotateY(spinY));

mv1 = mult(mv, translate(0.0, 0.0, -0.02));
mv1 = mult(mv1, scale(1.6, 1.0, 0.1));
gl.uniformMatrix4fv(matrixLoc, false, flatten(mv1));
gl.drawArrays(gl.TRIANGLES, 0, numVertices);

mv1 = mult(mv, translate(0.0, -0.6, 0.0));
mv1 = mult(mv1, scale(0.1, 0.3, 0.05));
gl.uniformMatrix4fv(matrixLoc, false, flatten(mv1));
gl.drawArrays(gl.TRIANGLES, 0, numVertices);

mv1 = mult(mv, translate(0.0, -0.8, -0.05));
mv1 = mult(mv1, scale(0.8, 0.1, 0.5));
gl.uniformMatrix4fv(matrixLoc, false, flatten(mv1));
gl.drawArrays(gl.TRIANGLES, 0, numVertices);
```

4. [Byggt á prófdæmi frá 2021] Leiktækið snúningsvegasalt er eins og venjulegt vegasalt (seesaw), nema að það getur líka snúist í hringi (þ.e. 360° um miðjuna). Þið eigið að skrifa forrit fyrir þrívíðri útfærslu á slíku leiktæki. Hjá okkur samanstendur það af þremur einingum: flatur stallur sem situr á jörðinni og veitir stöðuleika, miðsúla, sem stendur uppúr stallinum, og loks vogarstöng, sem situr ofaná miðsúlunni. Endar vogarstangarinnar geta farið upp og niður eins og í venjulegu vegasalti, en svo getur hún líka snúist um miðju sína (sem situr ofan á miðsúlunni). WebGL forritið ykkar á að láta útbúa þetta leiktæki og að láta það vera á hreyfingu, þ.e. vega salt og snúast á sama tíma. Skilið skjáskoti og hlekk á forritið ykkar.

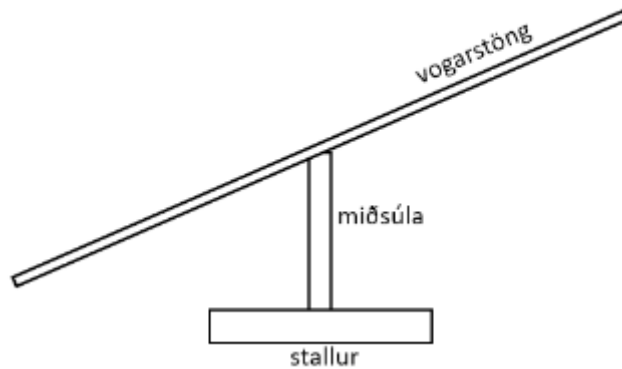


Figure 3: Snúningsvegasalt

Hlekkur



Figure 4: Skjáskot: Snúningsvegasalt

5. Hér sýnisforritið **solkerfi-tungl** sem hermir sólin, jörðina og tunglið. Þið eigið að bæta við gervihnetti sem snýst um tunglið. Tunglið snýst um jörðina í sama fleti og jörðin snýst um sóluna, en gervihnötturinn fer yfir póla tunglsins og hallar því 90° miðað við hinar sporbrautirnar. Bætið einnig við þeim eiginleika að upp/niður örvalyklarnir auka/minnka hraða hermunarinnar. Skilið skjáskoti og hlekk á forritið ykkar.

Hlekkur

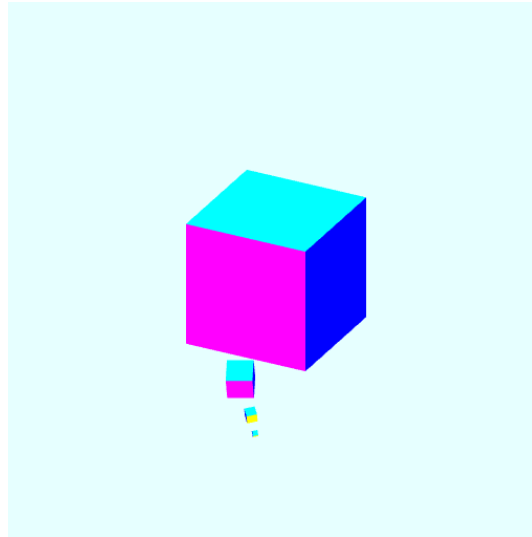


Figure 5: Skjáskot: Gervihnetti

```
window.addEventListener('keydown', function (e) {
  switch (e.keyCode) {
    case 38: // Up
      rotDayInc *= 1.05;
      rotYearInc *= 1.05;
      break;
    case 40: // Down
      rotDayInc *= 0.95;
      rotYearInc *= 0.95;
      break;
  }
});

rotDay += rotDayInc;
rotYear += rotYearInc;

// teikna "gervihnetti"
var mvt = mult(mvt, rotateZ(30.0 * rotYear * 2));
mvt = mult(mvt, translate(1.8, 0.0, 0.0));
mvt = mult(mvt, scalem(0.5, 0.5, 0.5));
gl.uniformMatrix4fv(matrixLoc, false, flatten(mvt));
gl.drawArrays(gl.TRIANGLES, 0, numVertices);
```