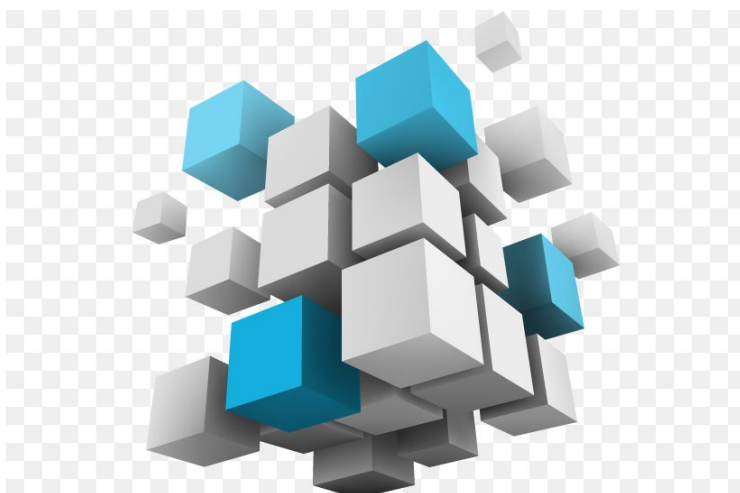


# Tölvugrafík Lokaprófa undirbúningur

brj46

Nóvember 2024



## Hvað þarf Ég Að Kunna?

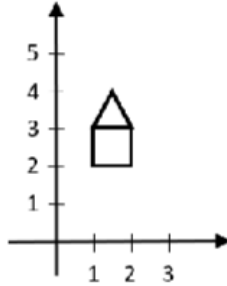
- ☐ Varpanir (tvívíðar og þrívíðar) (töluleg rök)
- ☐ Hreyfanleg líkön (WebGl) (Render-Föll)
- ☐ Endurskinslíkön (Phong lýsingarlíkanið) (útreikningar með dofnun)
- ☐ Mynsturvörpun (Texture mapping t.d. repeat og CLAMP\_TO\_EDGE) Ójafnhliða síun
- ☐ GLSL og litaraforritun (Hnúta- og bútalitarar) (Hreyfing og litabreytingar)
- ☐ Ýmis atriði í tölvugrafík (Homogeneous hnit) (þríhyrnaröðun)(Skuggakort)
- ☐ Aukaverkefni (Geislasmölun vs Phong) (Dofnun og birtustýring)

# Mín lausn við prófinu 2023

## Varpanir

a.

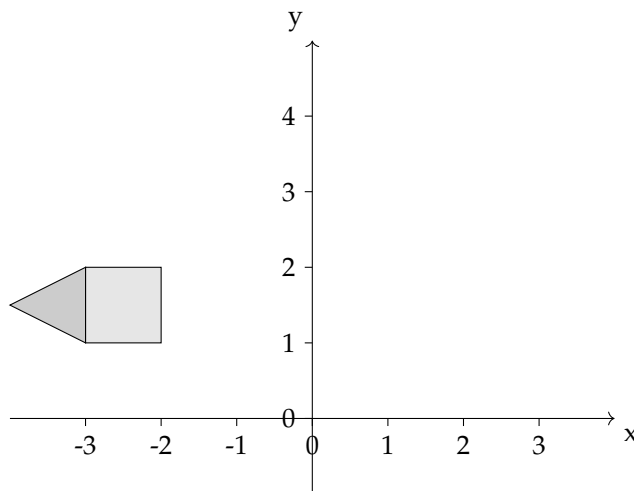
Á myndinni er gefin upphafsstaða á húsi í tvívídd. Neðra vinstra horn þess er í (1, 2).



Teiknið mynd af stöðu hússins eftir þessar samsettu varpanir (hvor liður sjálfstæður). Rökstyðjið í nokkrum orðum hver útkoman er.

i.  $T(1, 2) * S(1, 2) * T(2, -1) * R(90^\circ)$

- **Svar mitt** Alright skoðum hverja skipun fyrir sig (lesum frá hægri til vinstri)  $R(90^\circ)$  mun snúa húsinu um  $90$  gráður en frá  $0.0$  og snúið til vinstri svo næsta staða hússins væri



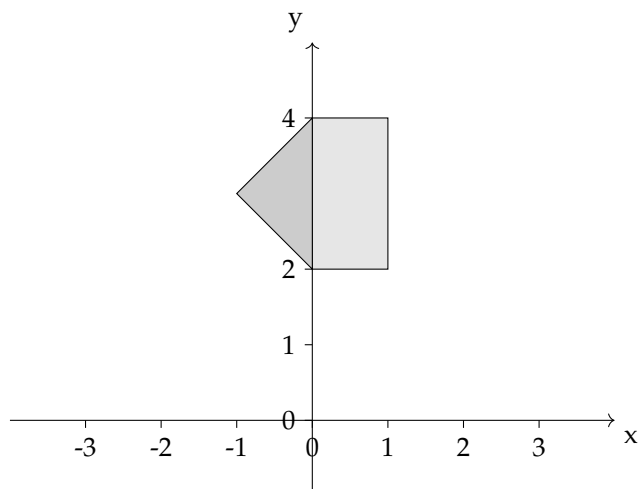
2.  $R(90^\circ)$

Næst er  $T(2, -1)$  þá munum við færa húsið 2 til hægri og 1 niður.

Svo er skölunin  $S(1, 2)$  sem mun skala húsið með 1 í x-ás og 2 í y-ás. semsagt tvöfaldast upp og viðheldur breiddinni.

Síðast er  $T(1, 2)$  sem færir húsið 1 til hægri og 2 upp.

að lokum höfum við



ii.  $T(1,2) * R(90^\circ) * S(1,2) * T(-1,-2)$

- **Svar mitt** Jæja taka tvö ég hafði gert fyrra dæmið í heimdæmi svo ég var svoldið með það skulum gera þetta núna

Sama dæmi byrjum á að lesa frá hægri til vinstri.

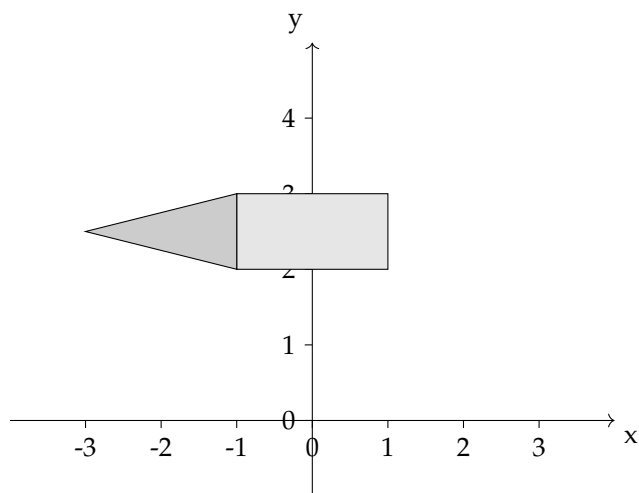
$T(-1, -2)$  færir húsið 1 til vinstri og 2 niður.

$S(1, 2)$  skalar húsið með 1 í x-ás og 2 í y-ás.

$R(90^\circ)$  snýr húsinu um 90 gráður til vinstri muna það því mér þykir það weird að snúa húsinu til vinstri.

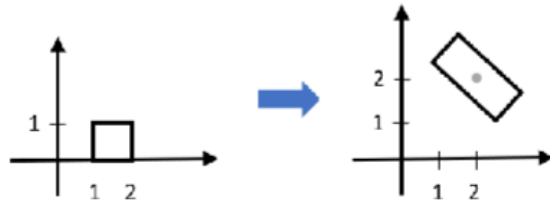
Síðast er  $T(1, 2)$  færir húsið 1 til hægri og 2 upp.

að lokum höfum við



b.

Sýnið samsetta tvívíða vörpun sem breytir feringnum í fyrri myndinni yfir í ferhyrninginn í seinni myndinni. Seinni ferhyrningurinn er með miðpunkt  $(2, 2)$ , hliðarlengdir 1 og 2, og halli hans er  $45^\circ$ . Rökstyðjið einstakar grunnvarpanir í samsettu vörpuninni.



## 2

### WebGL Útfærsla á líkani

Það er til útgáfa af klukku (Continue Time, hönnuður Sander Mulder) sem hefur þrjá arma eins og venjuleg klukka, en í stað þess að armarnir snúist allir um sama punkt (þ.e. miðpunkt skífu), þá eru þeir festir á endann á öðrum armi. Klukkustundararmurinn er reyndar festur á miðpunkt skífu og snýst um hann eins og á venjulegri klukku, en mínútuarmurinn er festur á endann á klukkustundararminum og snýst um þann punkt. Sömuleiðis er sekúntuarmurinn festur á endann á mínútuarminum og snýst um hann. Á myndinni hér til hliðar er klukkustundararmurinn á milli 4 og 5, mínútuarmurinn sýnir 30 og sekúntuarmurinn vísar á 45. Klukkan hér er því 4:30:45.



Sýnið uppkast að render-falli sem teiknar þessa klukku. Hver armur á að vera teningur og þið getið gert ráð fyrir að sé búið að hlaða hnúta hans inn í grafíkminni. Vísarnir eiga að liggja í xy-sléttunni og þeir eiga að snúast rétt, þannig að fyrir hvern heilan hring sekúntuarmsins snýst mínútuarmurinn sem svarar einni mínútu (þ.e. 1/60 af heilum hring). Samsvarandi gildir með klukkustundararminn. Þið eigið bara að uppfæra sekúntuarminn um eina sekúntu í hverri ítrun render-fallsins (og hina armana samsvarandi). Þetta gefur auðvitað ekki réttan tíma, en sýnir hreyfingu klukkunnar. Einbeitið ykkur að því að sýna varpanirnar og teikniföllin í render-fallinu. Þetta þarf ekki að vera alveg keyranlegur kóði, heldur skiptir meira máli að þið séuð að hugsa varpanirnar rétt. Útskýrið þess vegna vel einstakar skipanir.

#### Svar:

```
1 function render()
2 {
3     gl.clear( gl.COLOR_BUFFER_BIT | gl.DEPTH_BUFFER_BIT );
4
5     var proj = perspective( 70.0, 1.0, 0.2, 10.0 );
6     gl.uniformMatrix4fv(proLoc, false, flatten(proj));
7
8     var mv = lookAt( vec3(0.0, 0.0, zDist), vec3(0.0, 0.0, 0.0), vec3(0.0,
9     1.0, 0.0) );
10    mv = mult( mv, rotateX( spinX ) );
11    mv = mult( mv, rotateY( spinY ) );
12
13    // Ba til bakgrunn fyrir klukku
14    mvb = mult( mv, translate( 0.0, 0.0, -0.04 ) );
15    mvb = mult( mvb, rotateY( 180.0 ) );
16    mvb = mult( mvb, scalem( 1.25, 1.25, 0.01 ) );
17    gl.uniformMatrix4fv(mvLoc, false, flatten(mvb));
18    gl.drawArrays( gl.TRIANGLES, 0, NumVertices );
19
```

```

20
21     sek += sekupph;
22     sekvisir = 6*sek;
23     minvisir = 0.1*sek;
24     kstvisir = 0.1*sek/11.0;
25
26
27     // ----- Klukkustundavísir
28     mv = mult( mv, rotateZ( -kstvisir ) );           // Sna samkvmt
29     breytingu tma
30
31     // Líkanavörpun klukkustundavísir
32     mvk = mult( mv, translate( 0.0, 0.1, 0.0 ) );   // Hlira upp, svo
33     neri endi (0, 0, 0)
34     mvk = mult( mvk, scalem( 0.025, 0.2, 0.025 ) ); // Setja rtta
35     str
36
37     gl.uniformMatrix4fv(mvLoc, false, flatten(mvk));
38     gl.drawArrays( gl.TRIANGLES, 0, NumVertices );
39
40     // ----- Mínútuvisir
41     mv = mult( mv, translate( 0.0, 0.2, 0.0 ) );   // Fra t enda
42     klukkustundavísir
43     mv = mult( mv, rotateZ( -minvisir ) );           // Sna samkvmt
44     breytingu tma
45
46     // Lkanavrpun mntuvsis
47     mvm = mult( mv, translate( 0.0, 0.1, 0.0 ) );   // Hlira upp, svo
48     neri endi (0, 0, 0)
49     mvm = mult( mvm, scalem( 0.02, 0.2, 0.02 ) );   // Setja rtta str
50
51     gl.uniformMatrix4fv(mvLoc, false, flatten(mvm));
52     gl.drawArrays( gl.TRIANGLES, 0, NumVertices );
53
54     // ----- Seknduvsir
55     mv = mult( mv, translate( 0.0, 0.2, 0.0 ) );   // Fra t enda
56     mntuvsis
57     mv = mult( mv, rotateZ( -sekvisir ) );           // Sna samkvmt
58     breytingu tma
59
60     // Lkanavrpun seknduvsis
61     mvs = mult( mv, translate( 0.0, 0.1, 0.0 ) );   // Hlira upp, svo
62     neri endi (0, 0, 0)
63     mvs = mult( mvs, scalem( 0.015, 0.2, 0.015 ) ); // Setja rtta
64     str
65
66     gl.uniformMatrix4fv(mvLoc, false, flatten(mvs));
67     gl.drawArrays( gl.TRIANGLES, 0, NumVertices );
68
69     requestAnimationFrame(render);
70 }

```

### 3.

#### Endurskinslíkön

Hér fyrir neðan er líkan með áhorfanda, ljósgjafa og glansandi yfirborði. Áhorfandinn og ljósgjafinn eru í sömu hæð og beint fyrir ofan sitt hvort brotið í yfirborðinu. Gerið ráð fyrir að notað sé endurskinslíkan Phong án dofnunar til að búa til lit á yfirborðið.



a.

Hvar á yfirborðinu er bjartasta dreifendurskin (diffuse reflection)? Ef fleiri en einn staður eru jafnbjartir tilgreinið þá alla björtustu staðina. Rökstyðjið svarið í nokkrum orðum.

b.

Hvar á yfirborðinu er bjartasta depilendurskin (specular reflection)? Ef fleiri en einn staður eru jafnbjartir tilgreinið þá alla björtustu staðina. Rökstyðjið svarið í nokkrum orðum.

c.

Ef hægt væri að færa ljósið til hliðar (í sömu hæð), eru staðsetningar á ljósinu sem gefa ekkert depilendurskin á þessu yfirborði? Gerið ráð fyrir að glansstuðull yfirborðsins sér hár og depillinn sé því lítill. Rökstyðjið svarið.

d.

Ef ekkert umhverfisendurskin (ambient reflection) væri í líkaninu, væri þá einhver hluti yfirborðsins alveg svartur? Rökstyðjið.



#### 4.

Ýmislegt efni

Hér fyrir neðan eru stuttar spurningar úr ýmsu efni námskeiðsins. Svarið hverri spurningu með nokkrum setningum

**a.**

Berið saman Gouraud litun og Phong litun. Nefnið einn kost og einn galla við hvora aðferð.

**b.**

Er hægt að útfæra mynsturvörpun eingöngu í hnútalitara? Rökstyðjið svar ykkar.

**c.**

Skuggakort (shadow maps) eru ein leið til að fá skugga í WebGL. Lýsið stuttlega hvernig hún virkar og hvaða takmarkanir eru á aðferðinni.

**d.**

Útskýrið hvaða kosti það hefur að nota jafnbætt (homogeneous) hnit í tölvugrafík.

**e.**

Hvers vegna skiptir máli að röð hnúta í þríhyrningi sé rétt (þ.e. rangsælis eða réttsælis)? Rökstyðjið.

5.

### Mynsturvörpun

a.

Spurningar um ójafnhliða síun (anisotropic filtering):

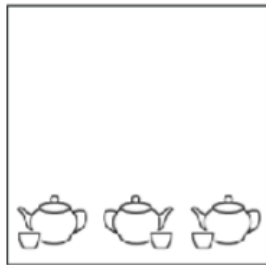
- i. Hvers vegna er ójafnhliða síun betri en "trilinear"síun (þ.e. MIP vörpun með **LINEAR-LINEAR** stillingu) á gólfi sem teygir sig langt frá áhorfanda? Rökstyðjið.
- ii. Eru einhver tilvik þar sem ójafnhliða síun gæfi verri útkomu en "trilinear"síun? Rökstyðjið.

b.

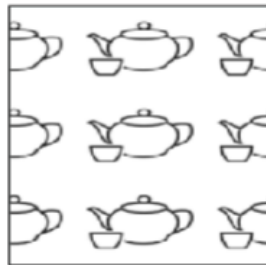
Hér fyrir neðan eru þrjár myndir af spjaldi þar sem búið er að varpa á það mynstrinu af tepottinum hér til hægri. Í hverju tilfelli gefið mynsturhnitin sem þarf að nota og tiltakið þær stillingar á mynsturvörpuninni sem skipta máli til að fá þessa útkomu.



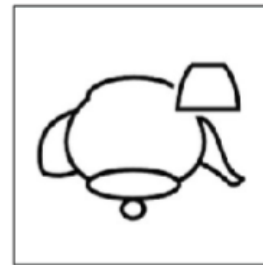
Mynstur



i.



ii.



iii.

## 6.

### Bakhliðareyðing og litaraförritun

#### a.

Segjum að við litum margflötung (polytope) með endurskinslíkani Phong og Phong litun. Ef það er ljósgjafi fyrir aftan margflötunginn (frá áhorfanda) og ef bakhliðareyðing (back-face culling) er ekki virk, hvernig lit fá þá bakhliðar margflötungsins (ef við sæjum inn í hann)? Rökstyðjið.

#### b.

Hvernig breytist svar ykkar í a-lið ef bakhliðareyðing er virk? Rökstyðjið.

#### c.

GLSL bútalítarar hafa innbyggðu breytuna **gl\_FrontFacing**. Þetta er breyta af taginu bool sem er sönn ef núverandi bútur er hluti af þríhyrningi sem snýr að áhorfandanum, en ósönn annars. Skriðið GLSL kóðabút sem veldur því að allar bakhliðar sem sjást eru rauðar, en framhliðar væru litaðar eins og venjulega (þið þurfið ekki að skrifa út þann hluta kóðans!). Hvað myndi gerst í kóðanum ykkar ef bakhliðareyðing væri virk? Útskýrið.