**3D Modellezés és Animáció tematika összefoglaló**

**Kamera nézetek:**

* axonometrikus: nem perspektív, megőrzi a párhuzamosságot
* perspektív: 2 vagy 3-pontos perspektíva
* kamera: egy kamera nézetére állítja a kamerát
* fény: fényforrásra állítja a kamerát

**Koordináta-rendszerek:** Transzformációknál használatos (scale, move, rotate).

* View: XYZ a kiválasztott nézet szerint (X jobbra nő, Y felfelé, Z kifelé)
* Screen: XYZ a képernyő szerint (X jobbra nő, Y felfelé, Z kifelé)
* World: XYZ a világ koordinátarendszer szerint
* Parent: A szülő koordinátarendszerét használja (root parent a World)
* Local: A pivot point koordinátarendszerét használja
* Gimbal: Kontrollereknél hasznos, motion trackhez képest vett koordináták
* Grid: Az aktív gridhez képest
* Working: Kiválasztott working pivot pointhoz képest
* Pick: Egy kiválasztott object koordinátarendszerét használja

**Bool műveletek:**

* Összeadás: Két objektumot egybeveszi.
* Kivonás: Két objektum közös metszetét veszi.

**Modifierek:**

* Taper: Egyik oldalt összenyom, másik oldalt széthúz
* Bend: Elhajlítja az objectet.
* Twist: Megcsavarja az objectet.
* Edit Mesh: Hozzáférést ad a vertex, edge, face/poly/element módosításokhoz (nem animálható)
* Edit Spline: Hozzáférést ad a vertex, segment és spline módosításokhoz (nem animálható)
* Lathe: Alakzatot tengely körül megforgatva hengerszimmetrikus alakzatot hoz létre.
* Extrude: Kiválasztott dolog objektumhoz képest vett mélységét változtatja (benyom, kihúz, stb.)
* Loft: Objektum keresztmetszetét állítja be egy shape-re.
* Mirror: Tükrözi az objektumot egy síkra (gizmoval állítható).

**Kamerák:**

* Target: A kamera és a target külön animálható, egyszerűbb, ha a kamerának mozognia kell.
* Free: A kamera arra néz, amerre forgatjuk, nehéz ha a kamera is mozog, nem csak amit filmez.

**Fények:**

* Ambient: Nincs forrása, vagy iránya, mindent egységesen megvilágít mindenhol.
* Omni: Egy pontból kiinduló, minden irányba terjedő fény.
* Spotlight: Egy pontból kiinduló, egy irányba terjedő fény (cone).

**Basic anyagparaméterek:**

* Ambient: Szín árnyékban.
* Diffuse: Szín direkt fényben.
* Specular: Csillogó felületeken a csillogás színe.
* Self Illumination: Az árnyékos részeket Diffuse-ra színezi, nem Ambient-re.
* Highlight: Specular a csillogás intenzitása, Glossiness a mérete (ha nő, a méret csökken), Soften a csillogás széleinek elsimítása. A harang ennek megfelelően Specular – magasság, Glossiness – szélesség (fordítottan arányosan), Soften – lekerekítettség

**Mappolás:** Anyagok realisztikussá tételében van szerepe.

* Diffuse, Ambient, Specular, Glossiness, Self-illumination: ugyanaz, mint fent, csak textúraként
* Opacity: Átlátszóság.
* Bump: Felületi érdesség, egyenetlenségek.
* Reflection: Tükröződési képesség, mintha a felszínről tükröződne.
* Refraction: Tükröződési képesség, mintha átlátnánk a felületen.
* Displacement: Hasonló mint a bump-map, csak ez a tényleges geometrián módosít, nem csak úgy néz ki mintha.
* Filter color: Például festett üvegen átvilágítunk lámpával, és az kivetíti a mintát.

**Procedurális mapok:** Nem pixeles kép, hanem algoritmus által generált.

* Checker: Sakktábla minta.
* Márvány: Márvány minta.
* Fa: Fa minta.
* Raytrace: Raytracelt tükröződés, pontosabb, mint a Reflect/Refract mapok.
* Flat mirror: Alkalmasabb tükörnek, mint a Reflect/Refract mapok.

**UVW mapok:** Bitmapokat a targetre pozícionál, UVW megfelel XYZ koordinátáknak.

* Planar: Egy síkot képez rá az objektumra.
* Cylindrical: Egy hengerként képez rá az objektumra.
* Cap: Olyan mint a cilinderes, csak az alapok planárisan leképezettek.
* Spherical: Egy gömböt képez le, úgy mintha körbe-ölelné egy palásttal, szingularitása van a pólusoknál.
* Shrink wrap: Ugyanúgy spherical projection, de csak egy szingularitása van („labda papírzacskóba csomagolva”)
* Box: Négyzetes doboz, 6 oldalról projectel.
* Face: Minden face-re ráteszi a textúrát.

**Egyéb anyagtípusok:**

* Standard: A legegyszerűbb, magától értetődő anyag.
* Multi-subobject: Több anyagot lehet belehúzni slot-okba, és az sub-objectek anyagai ezek száma alapján választhatók egy objektumnál.
* Blend: Összekever két anyagot úgy, hogy interpolál a kettő között.

**Szelekció:**

* Bekerítéses: Téglalap, kör, fence (sokszög), lasszó (rajzolt alakzat), paint (befestéses)
* Név szerint
* Selection sets: Nevesített kijelölési csoportok.
* Selection filters: Csak kategóriákat enged kijelölni (pl. fények, kamerák, formák, geometriák)

**Transzformációs középpontok:**

* Move: Mozgatás középpontja
* Rotate: Forgatás középpontja
* Scale: Átméretezés középpontja

**Scale fajták:**

* Uniform: Minden tengely mentén.
* Non-uniform: Csak egy tengely mentén.
* Squash: Több tengely mentén.

**Modifier stack felépítése:**

* creation paraméterek
* modifierek
* transzformációk
* térgörbítők

**Stack bejegyzések számossága:** Memóriamegtakarítási megfontolások miatt véges, alapértelmezés szerint 1, a legutoljára használt módosító(k) eredményét a program eltárolja, hogy újabb módosításnál gyorsabban meglegyen a végeredmény.

**Sub-object:** Egy geometria részei, részhalmazai, pl. objektum bizonyos vertexei, élei, oldalai, stb. Az Edit Mesh modifierrel, vagy Editable Mesh-é konvertálással szerkeszthetőek.

**Modifier gizmo:** Ha a módosító térbeli pozícionálásra érzékeny, a gizmo segítségével tudjuk ezt beállítani.

**Objektumok felépítése:**

* Vertex: Térbeli pont, csúcs
* Edge: Él, csúcsokat köt össze
* Face: Egy oldallap, egy síkban lévő csúcsok és élek határolják
* Element: Összefüggő face-ekből álló geometria
* Object: Az elementek összessége (ha az egész geometria összefüggő az object = element)
* Group: Objektumok csoportja

**Térgörbítők:** Az objektumok geometriáját módosítják, vagy erőtereket modelleznek (pl. gravitáció), amik animációknál hasznosak. Két fajtájuk van, a térgörbítő objektumok és a modifierek. A kettő között az a különbség, hogy a modifierek csak az adott objektumra vannak hatással, amire rátettük őket, míg a térgörbítő objektumok a világ részei, és minden más objektumra hatással lehetnek, amikhez hozzárendeljük őket.

**Stack végrehajtási sorrend:**

* Creation paraméterek
* Modifierek
* Transzformációk (move, scale, rotate)
* Térgörbítők
* Objektum-tulajdonságok (pl. alkalmazott material)

**Morfózis:** Egy geometria átalakul egy másik geometriává, például: ember átalakul állattá. A morfózis feltétele, hogy a két geometriának azonos számú vertex-el kell rendelkeznie.

**Másolás:**

* Copy: Teljesen különálló másolat, a teljes stack duplikálódik.
* Instance: Az eredetivel összefüggő másolat, bármelyiket változtatjuk, a másik is változik. A creation paraméterek és modifierek közösek.
* Reference: Hasonló az Instance-hez, de csak a másolás pillanatában fennálló tulajdonságaik maradnak közösek (creation paraméterek, modifierek), az utólagosan alkalmazott változtatások (új modifierek), transzformációk és térgörbítők viszont egyediek.

**Összetett másolás:**

* Array: Koordinátatengelyek mentén, megadható számú másolatot hoz létre, megadható távolságra egymástól.
* Spacing tool: Egy görbe mentén helyezi el a másolatokat.
* Snapshot: Mozgó objektumról snapshot másolatokat készít, egyes időpillanatokban.

**Animation kontrollerek:** Animációkat irányítanak, kulcs értékeket lehet velük beállítani bizonyos időpontokra, amik között interpolációt végeznek.

* Bezier: Két kulcs között matematikai görbével megadható sebességgel interpolál.
* Path constraint: Az objektum mozgását egy alakzat mentén való mozgásra korlátozza.
* Look at constraint: Az objektum mindig egy másik objektum felé néz.
* Link constraint: Az objektum örököli a target objektum transzformációit.
* Expression: Az objektum tulajdonságait egy matematikai függvény határozza meg.
* Noise: Az objektum tulajdonságait random értékre állítja, amik időben változnak (zaj).

**Out-of-range típusok:** Meghatározzák, hogy egy objektum hogyan viselkedik az animáció során azokban a pillanatokban, amiket kulcsokkal nem határoztunk meg (pl. +végtelenhez vagy –végtelenhez tartva időben)

* Constant: Az adott tulajdonság konstans az animáció adott időtartama alatt.
* Cycle: Az animációt ismételgeti, de a tulajdonság az elején nem ugyanaz, mint a végén.
* Loop: Ugyanaz, mint a Cycle, csak elsimítja az eleje és a vége közötti különbséget.
* Ping-pong: Ismételgeti az animációt oda-vissza.
* Linear: Lineárisan játssza le az animációt.
* Relative repeat: Ismételgeti az animációt, de minden ismétlés között vár egy meghatározott ideig.

**Pivot point:** A transzformációk (move, scale, rotate) középpontja, a modifierek alapértelmezett középpontja, a linkelt child objektumok transzformációjának originje.

**Forward kinematika:** A linkelés alapja, hogy ha a szülő mozog, a gyerekek vele mozognak, de ha a gyerekek mozognak, a szülő egy helyben marad. Például, ha egy ember előre hajol, a feje is előrébb kerül, de ha csak a nyakát mozgatja, akkor nem dől előre a teste.

**Link inheritance:** Alapértelmezés szerint a gyerek objektum örököli a szülő objektum transzformációit. Ez a viselkedés felülírható, az egyes transzformáció-fajták (move, scale, rotate) öröklésének letiltásával illetve engedélyezésével.

**NURBS modellezés:** Az objektumot nem szögletes alakzatokból képezzük, mint a hagyományos vertex-poly modellezésnél, hanem matematikai görbékből, ezért nagyon könnyen lehet vele komplex alakzatokat modellezni.

**Patch:** Ha egy objektumot patch-é konvertálunk, a program minden élét Bezier-görbékké alakítja, így könnyen lehet finoman ívelt formákat modellezni vele.

**MassFX:** 3D fizikai motor, valósághű fizikai szimulációkhoz.

**Részecske rendszerek:** Részecskéket (geometriai alakzatokat) spawnolnak. Segítségükkel különböző effekteket hozhatunk létre, pl. tűz, fröccsenő víz, füst, stb.

**Script:** Szkriptelni lehet vele bazd meg, mire lenne jó, Istenem?

**Video Post:** Animáció utáni videós utómunkákat lehet vele csinálni, mint pl. vágás, videók egymásra illesztése, stb.

**Camera Match:** Segítségével létrehozhatunk egy program-beli kamerát egy fotó alapján úgy, hogy a kamera pontosan azon a helyen van, és olyan tulajdonságokkal rendelkezik, mint a fotót készítő valóságos kamera.

**Phong modell:** Phong visszaverődés = Ambient + Diffuse + Specular

**Bump-map:** Fények számolásakor a tényleges felületet nem módosítjuk mint Displacement-nél, hanem egy ideiglenes felületet hozunk létre amit aztán a visszaverődések számításakor használunk. Sokkal gyorsabb, mint a Displacement, mert a tényleges felületet geometriailag nem változtatjuk. Módszer: A bump-mapből számított felületi normálist interpoláljuk az eredeti felület normálisával, és az árnyalás számításakor ezt az interpolált felületi normálist használjuk.

**Anti-aliasing:** Egy grafikában használatos technika, több fajtája létezik, célja, hogy a modellben lévő éles kontúrokat elsimítja, így az realisztikusabban néz ki. Hatása akkor látszik a leginkább amikor egy renderelt él görbe, vagy ferdén áll és egy olyan képernyőn jelenítjük meg amin a pixelek függőlegesen és vízszintesen helyezkednek el egymás mellett. Anti-aliasing nélkül ezek éles kontúrokkal lépcső-szerűen rajzolódnának ki, így viszont ezek az élek elsimulnak.