

Domaći - izometrije prostora

Prvi zadatak: Implementirati funkcije 1-6. Svaka nosi po 2 boda.

Test primer da koristi redom funkcije. Prve 4 funkcije: Treba da krene od neka 3 Ojlerova ugla, za njih izračuna matricu A , nju da predstavi preko ose i ugla, za dobijenu osu i ugao da odredi matricu A i na kraju za matricu A da dobije njene Ojlerove uglove. Oni bi trebali biti (osim u "gimbal lock" slučaju) isti kao uglovi od kojih ste krenuli.

5. i 6. funkcija: Na osnovu rotacije preko ose i ugla koju ste prethodno koristili, dobijete jedinični kvaternion, a zatim ga vratite u (trebalo bi opet istu) osu i ugao.

Kod (samo implemenaciju funkcija) mi šaljete kao domaći zadatak u Moodle sistemu. U fazi razvoja možete koristiti test primer ~~Testera Testeroića~~. Od njega različit test primer, tj. rezultate svake funkcije, ~~mi~~ šaljete kao snimak dela ekrana u okviru domaćeg zadatka. ~~To što pošaljete je vidljivo samo meni.~~

1. `Euler2A` $[\phi, \theta, \psi]$ - vraća matricu $A = R_Z(\psi)R_Y(\theta)R_X(\phi)$.
2. `AxisAngle` $[A]$ - vraća jedinični vektor $p = (p_x, p_y, p_z)$ i ugao $\phi \in [0, \pi]$ tako da $A = R_p(\phi)$.
3. `Rodriguez` $[p, \phi]$ - vraća matricu rotacije oko orjentisanog (jediničnog) vektora p za ugao ϕ .
4. `A2Euler` $[A]$ - za datu ortogonalnu matricu A , $\det A = 1$, vraća Ojlerove uglove ϕ, θ, ψ , redom.
5. `AxisAngle2Q` $[p, \phi]$ - vraća jedinični kvaternion $q = (x, y, z, w)$ tako da $C_q = R_p(\phi)$. Vektor p je jedinični.
6. `Q2AxisAngle` $[q]$ - vraća jedinični vektor $p = (p_x, p_y, p_z)$ i ugao $\phi \in [0, \pi]$ tako da kvaternion predstavlja rotaciju $R_p(\phi)$, tj. $C_q = R_p(\phi)$.

~~**Drugi zadatak:** (nosi 8 bodova) Implementirati funkciju:~~

~~`slerp` $[q_1, q_2, t_m, t]$ - vraća jedinični kvaternion q koji predstavlja SLerp interpolaciju između kvaterniona q_1 i q_2 , u trenutku $t \in (0, t_m)$. Kvaternioni q_1, q_2 su jedinični (važi `slerp` $[q_1, q_2, t_m, 0] = q_1$; `slerp` $[q_1, q_2, t_m, t_m] = q_2$.)~~

~~Zatim napraviti animaciju koja radi sledeće: zadaje objekat (po vašem izboru) u dve položaja (zadata pozicijama centara i orijentacijama Ojlerovim uglovima). Interno prebacujete Ojlerove uglove u dva kvaterniona (jedan za svaki položaj objekta). Zatim pravite animaciju koja prevodi jedna objekat u drugi (linearno interpolira translaciju između centara objekata, a Slerp interpolira njihovu orijentaciju).~~

~~Animaciju kačite u okviru odgovarajućeg foruma, kod Slerp funkcije mi saljete preko domaceg (ne treba mi kod za grafičko okruženje, samu animaciju, ...). Animacija koju kačite u okviru posta mora da bude dovoljno mala memorijski. Ako želite neku koja lepše izgleda, tj. ima dosta Mb, postavite link u postu.~~