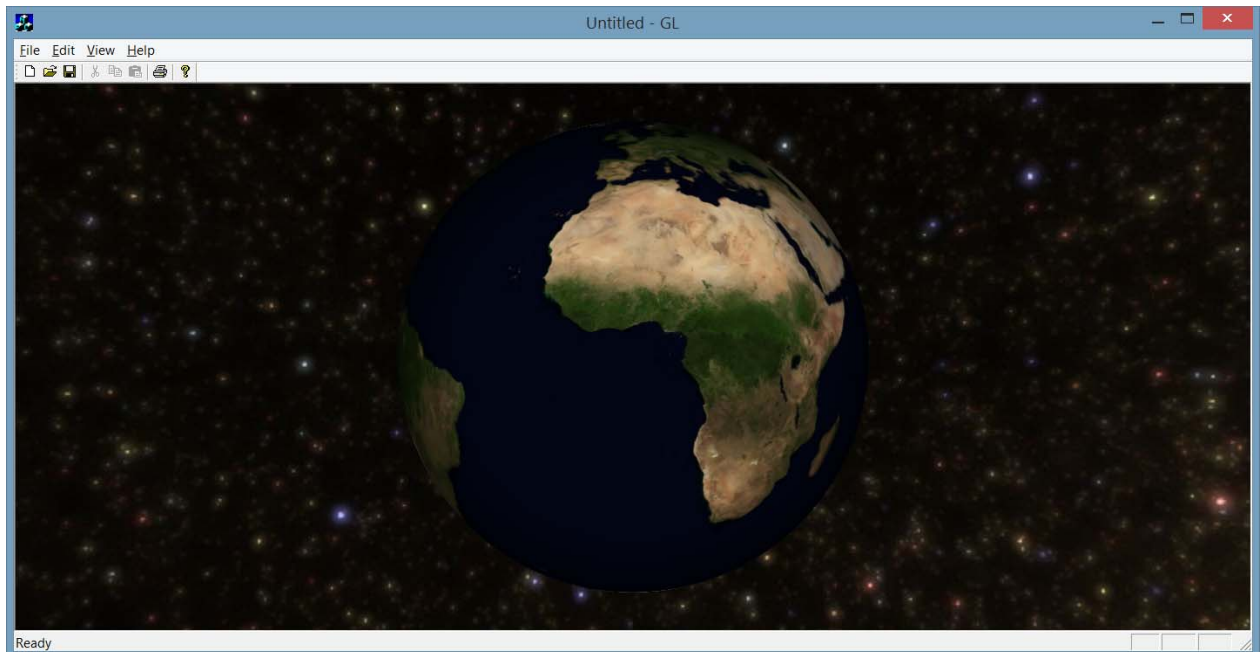


Računarska grafika

II kolokvijum

26.01.2015.



Sl. 1

1. Definirati perspektivnu projekciju sa $FOV = 40^\circ$ i ispuniti funkcije `PrepareScene()`, `DrawScene()` i `Reshape()` odgovarajućim OpenGL funkcijskim pozivima kako bi se omogućilo dalje crtanje. Napomena: Nacrtati trougao u koordinatnom početku i preći na sledeću tačku tek kada trougao bude vidljiv. [10 poena]
2. Napisati funkciju `GLuint CGLRenderer::LoadTexture(char* fileName)`, koja učitava teksturu sa datim imenom (`fileName`) i vraća ID kreirane teksture. [10 poena]
3. U funkciji `void CGLRenderer::PrepareScene(CDC *pDC)`, učitati teksture: `TSC0.jpg` – `TSC5.jpg`, `M0.jpg` – `M5.jpg`, `S0.jpg` – `S5.jpg`, i postaviti sve potrebne početne vrednosti parametara. Izvršiti dealokaciju učitanih tekstura u odgovarajućoj funkciji. [5 poena]
4. Napisati funkciju `void CGLRenderer::DrawPatch(double R, int n)`, kojom se iscrtava četvorostrana zakrivljena površina (*patch*) koja predstavlja 1/6 površine sfere (Sl.2). Patch je iscrtava parametarski po $x, y \in [-1, +1]$, pri čemu je n broj podeoka po pravcu na koliko je podeljen patch. Na Sl.2, $n = 20$. Iz parametarskih koordinata prelazi se u polarne po jednačinama:

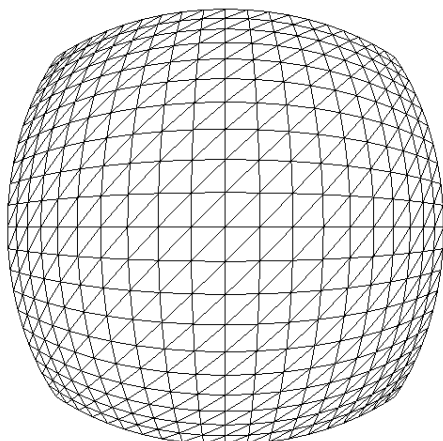
$$\begin{aligned}\varphi &= \tan^{-1} x \\ \theta &= \tan^{-1}(y \cdot \cos \varphi)\end{aligned}$$

a iz polarnih u Dekartove po jednačinama:

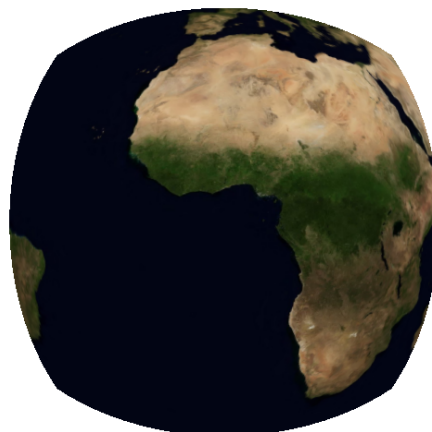
$$\begin{aligned}x &= R \cdot \cos \theta \cdot \sin \varphi \\ z &= R \cdot \cos \theta \cdot \cos \varphi \\ y &= R \cdot \sin \theta\end{aligned}$$

U temenima generisati teksturne koordinate i normale. Teksturne koordinate treba da omoguće mapiranje čitave teksture na čitavu površinu patch-a (Sl.3). Normale se postavljaju radijalno. [25 poena]

5. Korišćenjem funkcije DrawPatch() napisati funkciju `void CGLRenderer::DrawEarth(double R, int tes)`, koja icerava čitavu površinu Zemlje, sastavljenu od 6 patch-eva i na nju primenjuje teksture TSC0 – TSC5. Redosled primena tekstura prikazan je na Sl.4. [10 poena]
6. Na isti način napraviti funkcije DrawMoon() i DrawSpace(), ali u njima primeniti teksture M i S, respektivno. [5 poena]
7. Popuniti funkciju `void CGLRenderer::DrawScene(CDC *pDC)`, tako da u centru scene bude Zemlja (*Earth*), poluprečnika 3, na 50 jedinica od nje Mesec (*Moon*), poluprečnika 1, a centrirana u kameri bude sfera koja predstavlja Svemir (*Space*). [5 poena].
8. Postaviti direkcioni izvor svetlosti bele boje, koji se nalazi u beskonačnosti u pravcu pozitivne Z-ose. Izvor svetlosti ne sme da utiče na Svemir, niti da se pomera sa posmatračem. Uticaj svetla uključivati/isključivati na taster S. [10 poena]:
9. Omogućiti animiranje scene tako da pritisak na taster:
 - ← – rotira posmatrača oko Y-ose udesno oko Zemlje,
 - → – rotira posmatrača oko Y-ose ulevo oko Zemlje,
 - ↑ – rotira posmatrača naviše,
 - ↓ – rotira posmatrača naniže,
 - + – približava posmatrača centru Zemlje
 - – – udaljava posmatrača od centra Zemlje
 - Q/W – rotira Mesec oko njegove ose. [20 poena]



Sl.2



Sl.3

TSC4			
TSC0	TSC1	TSC2	TSC3
TSC5			

Sl.4