

Važnost "IKT" u metodici nastave na graf. fakultetu ; OSVRT

Što je uopće IKT? IKT odnosno informacijska i komunikacijska tehnologija, djelatnost i oprema koja čini tehničku osnovu za sustavno prikupljanje, pohranjivanje, obradu, širenje i razmjenu informacija različitih oblika.

U ovom videu „Važnost IKT u metodici nastave na grafičkom fakultetu“ prof. dr. sc. Klaudio Pap je na vrlo jasan i zanimljiv način objasnio upravo tu važnost. Započeo je sa objašnjavanjem fonta koristeći program "Fontographer". Font zapravo, u grafici, možemo definirati kao uređenu skupinu kodnih pozicija, a na svakoj kodnoj poziciji nalazi se slika (glif). Sama ova definicija, nekome tko se prvi put susreće sa radom u "Fontographeru", ne znači puno. No, kada je to prikazano u samom programu, stvar postaje jasnija. Naučila sam kako slovo uopće stoji u koordinatnom sustavu (digitalnom četvercu), kako od baznog slova (primjerice C) dobiti ć, č, kako možemo mijenjati našu sloganicu slova (približili smo slovo V slovu A). Sljedeća točka odnosno riječ je o simulatoru koji je napravljen upravo na grafičkom fakultetu. Radi se o softveru koji je napisan u C++ jeziku, a omogućuje nam da jednostavno i brzo stvorimo sliku od dva slova prikazana u "Photoshopu". Zanimljivo je koliko nam ovaj softver olakšava posao jer moguće je sam (ručno) dizajnirati ovu masku slova u "Illustratoru", no za to bi nam bilo potrebno 3-4 sata. Profesor Pap je pokazao te objasnio kako taj softver izgleda u postscript programu u obliku koda. Iako kodiranje, meni osobno, nije najzanimljivija stvar, cijela je stvar objašnjena na jako zanimljiv i razumljiv način. Nadalje smo imali primjer još jednog postscript softvera. Softver je stvorio spiralu slova koju, upravo preko njega, možemo sami dizajnirati mijenjajući postavke rezolucije (ovisi o kvaliteti tiska), visine fonta, spirale (faktor snage spirale) i dr. Ponovno smo vidjeli i kodni zapisi softvera te kako se svaka naredba zapisana kodno, zapravo već nalazi u softveru.

Nakon toga profesor Pap je objasnio važnost i samu definiciju Bézierove krivulje. Bézierove krivulje su parametarske krivulje III. stupnja koje u računalnoj ^(vektorskoj) grafici služe za predviđanje položaja tijela krivulje. Danas su Bézierove krivulje postale standard u "Fontographeru", "Illustratoru" i drugim programima u kojima se rade vektorske slike.

Problem je u tome što prilikom ispisa, ispisne tehnologije razumiju samo postscript jezik. Zbog toga su nam važni postscript driveri koji će bilo koji drugi format konvertirati u postscript. Sljedeća vježba je upravo vezana za ovo. Otvorili smo "Ghostscript" softver gdje bilo koji postscript kod koji u ps ili eps opciji pošaljemo, on će ga prikazati. Prikazano nam je kako u text editoru možemo utjecati na Bézierovu krivulju (hauku auta) preko naredbe CURVETO (naredba za Bézierovu krivulju). Na ovaj način moguće je redizajnirati bilo koji dio auta tj. nekog drugog objekta. Kao što je rekao i profesor, smatram da ćemo preko naredbi puno bolje naučiti bit Bézierovih krivulja. Naravno Bézierovu krivulju koristimo i u mnogim drugim jezicima i tehnologijama, no najviše se primjenjuje u računalnoj grafici i srodnim djelatnostima. Možemo je primjerice pronaći u vektorskoj grafici za WEB; SVG jeziku.

SVG jezik je jezik za prikazivanje dvodimenzionalne vektorske grafike, kao i standard ^{srodan} HTML i CSS. SVG je jako sličnih naredbi kao postscript. Mogli smo vidjeti kratku animaciju (kretanja trokuta po krivulji) napravljenu upravo u SVG-u. Koliko god povećavali animaciju, njena rezolucija ostaje ista. Što znači da vektorska grafika nije vezana za rezoluciju već samo za moment ispisa kada nešto prikazujemo. Zatim je objasnio kod za izradu ovakve animacije. Riječ je o malom kodu pomoću kojeg opet možemo mijenjati nama vidljive objekte animacije.

Primjerice možemo ukloniti stazu, promijeniti joj boju, ubrzavati animaciju i slično. Također je sve opet detaljno i jasno objasnio na primjeru torusa, gdje je program vrtio kružnicu po pethji od 0° - 360° . Zanimljivo je zapravo koliko se promjene može napraviti ako utječemo

na sam kod, a ne samo pomoću softvera. No za to je potrebno poznavati naredbe i matematičke formule.

Predzadnja točka je kontrola rastera odnosno rastriranje.

Rastriranje je napravljeno kako bi s jednom bojom čovjek mogao napraviti n-nijansi. Kod digitalnog rastriranja, rasterska ćelija se sastoji od matrice pixela. Na primjeru jedne slike mogli smo vidjeti amplitudno modelirano rastriranje gdje mi zapravo cijelo vrijeme imamo istu frekvenciju udaljenosti, ali se simulacija sivocije radi sa povećanim ili smanjenim rasterskim elementom (pixelom). Ovaj dio predavanja posebno mi se dojmio, posebno zato jer sam i sama mislila da je pixel tamnije boje ili svijetlije, ali naučila sam kako je to samo optička varka koja ovisi o gustoći točkica po inču. Također zanimljivo je kako u raster možemo podmetnuti različite funkcije (sinusoida).

Za kraj profesor Pop je rekao nešto i o digitalnoj boji. Naučila sam, zapravo ponovila, da se boja na ekranu (RGB sustav) razlikuje od one koju ispisujemo na naš proizvod (CMYK sustav). Potrebno je poznavati dobro oba sustava da bi znali barati bojom te njihovim značajkama.

Za kraj mogu reći da će mi ovaj video (predavanje) biti uvelike korisno za daljnji rad u grafičkoj tehnologiji.