

## Što je PostScript i što nam nudi?

### Što je PostScript?

PostScript je grafički jezik za opis stranice koji nam omogućava rješavanje složenijih grafičkih problema. Što to zapravo znači? Možda na prvu to tako ne izgleda, ali grafički prikazi na našim ekranima temelje se isključivo na skupu naredbi napisanih u obliku kodova. Naravno, ni to nije početak priče, ali s obzirom da se knjiga „PostScript“ ne bavi tom tematikom, neću ulaziti u detalje koji nisu spomenuti u navedenom djelu.

Do sada smo se već susreli s programima kao što su Illustrator i Inkscape, a oba programa nam omogućuju rad s tipografijom, bojama, raznim oblicima, linijama i sl. na način da odaberemo neku već unaprijed zadanu funkciju i primijenimo je na našem projektu. Zvuči vrlo jednostavno, no što ako odlučimo sami napisati te funkcije i na taj način utjecati na prikaz na ekranu? PostScript nam omogućuje da krenemo od nule i sami osmislimo i napišemo kod koji će odrediti izgled odabranih elemenata. Doduše, za razliku od programskih jezika kao što su Pascal ili C, PostScript se fokusira isključivo na oblikovanje grafike namijenjene tisku. To se odnosi na tipografiju, umetanje oblika nastalih kombinacijom ravnih i zakrivljenih linija, obradu piksel grafike (palette boja, osvjetljenost, zasićenost itd.) i sl.

PostScript se sastoji od staza, operatora (komandi) i operandi (parametara). Komande su pisane kurentnim i verzalnim slovima ili u obliku kratica ili u obliku punih riječi engleskog jezika. U većini slučajeva, komandama prethode parametri koji definiraju ili označavaju način na koji bi one trebale djelovati. Komentare pišemo nakon znaka za postotak koji najavljuju tekst nevezan uz napisani kod, tj. tekst koji nam može služiti kao podsjetnik pri razumijevanju koda, a ne želimo da utječe na rad cjelokupnog koda. Osim toga, vrlo je važno spomenuti i koordinatni sustav sastavljen od točaka s horizontalnom (x) i vertikalnom (y) koordinatom koje određuju njezinu poziciju.

Knjiga je podijeljena na tri poglavlja:

- 1) Programiranje grafike u kojem se upoznajemo s osnovama PostScripta te crtanjem ravnih i zakrivljenih linija, čijim modifikacijama ili ponavljanjem nastaju vrlo zanimljivi oblici (spominje se i Bezierova krivulja koja nam je već poznata s prošlog predavanja te programiranje triju tipova mreža koje čine podlogu)
- 2) Programiranje tipografije koja se temelji na izgledu znakova (oblik, ispuna i veličina slova, brojeva, interpunkcijskih znakova i sl., razmaci među njima itd.)
- 3) Programiranje piksel grafike koja se pretežno fokusira na obojenje piksela, ali i njihovu transformaciju zbog čega dolazi do rotacije, skaliranja i drugih vizualnih promjena na slikama

## Programiranje grafike

Prvo poglavlje započinje predgovorom i tekstom koji opisuje osnovne karakteristike PostScripta što sam već obuhvatila odgovarajući na pitanje „Što je PostScript?“. Nakon toga se susrećemo s nekih osnovnim komandama kao što su *moveto*, *lineto*, *stroke* i *showpage*. Prve tri navedene komande odnose se na stvaranje linije na određenoj poziciji u koordinatnom sustavu, dok *showpage* briše trenutne prikaze i izmjenom parametara priprema postojeću stranicu kako bismo mogli stvoriti novu. Također upoznajemo komande za oblikovanje i orijentaciju linija, tj. promjenu debljine linija, preusmjeravanje pružanja linije kako bismo stvorili zatvoreni lik, ispunu/obojenje linija, njihove kutove na mjestu presjeka i završetke, kontinuitet i isprekidanost pri čemu nastaju točkaste ili iscrtkane linije itd. Bezierova krivulja u PostScriptu se crta pomoću komande *curveto* koja traži upisivanje triju točaka koje definiraju krivulju, dok se kao prva točka uzima posljednja točka koju smo označili u prethodnoj radnji. Crtanje nam olakšavaju naredbe *repeat* i *for* ukoliko želimo nacrtati više istih objekata ili primijeniti iste naredbe nad uzastopnim elementima, *scale* ako im želimo mijenjati veličinu te *rotate* ukoliko ih želimo rotirati oko točke rotacije. Kako bi program upamtio određene vrijednosti, možemo koristiti komandu *gsave*, te

grestore u trenutku kada nam te vrijedosti postanu suvišne, tj. nepotrebne pa ih jednostavno izbrišemo. Te vrijednosti slažu se u stack tako da prvi član s desne strane niza predstavlja element na vrhu, a prvi član s lijeve strane predstavlja element na dnu. U prvom poglavlju spominje se još mnogo komandi pa tako i matematičke operacije koje se odvijaju na vrlo sličan način kao i u ostalim programskim jezicima. Zatim su predstavljene tri vrste mreža s kojima se možemo susresti u PostScriptu (postoji i četvrta vrsta, ali ona nije objašnjena u knjizi). Na kraju poglavlja javljaju se komande za podešavanje triju sustava boja: RGB, CMYK i HSB.

## **Programiranje tipografije**

Cijelo drugo poglavlje odnosi se na tipografiju, dakle oblik znakova i način povezivanja skupa znakova u cjelinu. Na samom početku se spominje pojam „četverac“ čija je uloga veoma važna jer je to prostor unutar kojega formiramo slova i ostale znakove. Navedena su neka pravila koja bi bilo poželjno poštovati prilikom izrade slovnihi znakova kao što je razina do koje se pružaju kurentna slova koja je prikazana u obliku postotka u odnosu na visinu četverca. Kako bi se program lakše snalazio prilikom upisivanja više slovnihi znakova (ili bilo kakvih drugih znakova), koristi se stringovima. To su setovi znakova koje mi prepoznamo kao riječi, rečenice, odlomke teksta i sl. (uključujući i razmake među riječima te interpunkcijske znakove). Procedure za poravnavanje teksta LTEKST (ulijevo), RTEKST (udesno), CTEKST (centrirano) i JTEKST (obostrano), temelje se na stringovima koje program „čita“ te brojanjem razmaka i duljina skupa uzastupnih slovnihi znakova (mi ih vidimo kao riječi) raspoređuje znakove na zadani način tako da mi na svojim ekranima vidimo skup znakova koji bismo definirali kao „čitljiv tekst poravnat na željeni način“. Osim toga, kada je to potrebno, koristi stringove i pri izvršavanju procedure CRODIJELI i LPRELOM koje omogućuju dijeljenje riječi na kraju rečenice po hrvatskom pravopisu tako što program „čita“ string, tj. neku riječ i traži najbolje pozicionirani samoglasnik uz marginu i nakon njega umeće crticu te ostatak riječi šalje u novi red.

Program PostScript nudi nam brojne opcije za uređenje znakova, kao što su definiranje debljine obruba, ispuna, skaliranje, rotacija i sl. što je naravno primjenjivo i na skupu znakova koji predstavlja neku cjelinu. U tom slučaju možemo mijenjati i razmake među znakovima te ih grupirati na razne načine. Spominju se i razni operatori koje PostScript nudi npr. relacijski, aritmetički, logički, stack i string operatori itd. Veoma su slični operatorima koje nude i ostali programski jezici.

## **Programiranje piksel grafike**

U ovom poglavlju sve se veže uz pojam „piksel“. To je najmanji grafički element na koji se može razložiti digitalizirana slika. Definiraju ga položaj na ekranu, nijansa boje i intenzitet osvijetljenosti. Svaki piksel može sadržavati samo jednu boju, no u kombinaciji s ostalim pikselima (koji sadže različite boje), gubimo dojam oštine na rubovima piksela. Prvi piksel na ekranu nalazi se u gornjem lijevom kutu, a ostali se nižu udesno pa prema dolje. Poslijednji piksel je smješten u donjem desnom kutu. Svaki piksel ima određeno mjesto unutar mreže na ekranu pa tako omogućuje ispravan i jasan prikaz slike. U knjizi se spominje problem s usklađivanjem prikaza pri promjeni dimenzija piksel matrica. Ako je matrica veća od ponuđenog niza piksela, dio podataka bit će izgubljen, tj. presječen na dijelu koji se ne uklapa po veličini, a ako je matrica manja, program će sam popuniti prazna mjesta ponavljanjem podataka s početka stringa. Pri kraju poglavlja, govori se o koordinatama slike na kojoj radimo te kutovima deformacije koji nam omogućuju promjenu veličine slike (skaliranje), rotaciju u tri dimenzije, zrcaljenje i sl. Tu može nastati problem, pogotovo u slučaju skaliranja jer nakon što jednom smanjimo sliku, ona će izgubiti na kvaliteti pri ponovnom povećavanju. Doduše, u slučaju da se bavimo izradom plakata, možemo povećati dimenzije piksela u otisku jer je osnovna namjena plakata uočljivost s veće udaljenosti gledišta, zbog čega nam detalji prikazani pikselima neće biti važni jer ih i onako nećemo moći vidjeti.