

# OSVRT NA PREDAVANJE

## PODSRIPT

PostScript je programski jezik kao i Basic, Pascal, C te ima sličnu strukturu komandi ali ipak, namijenjen je samo području oblikovanja grafike na stranicama za tisak. PostScript ima svoj specifični grafički rječnik koji je nastao razvojem računarske tipografije, fotosloga, i računarske reprofotografije.

Da bi se odredio grafički oblik slova ili slike koristi se komandni jezik kojim se određuje "staza" ili put te "operator" i "operandi" (parametri).

Koordinatni sustav PostScripta je zasnovan u mjernim veličinama "točka" sa X/Y koordinatama, a početak je u donjem lijevom dijelu stranice.

Program i grafike u prvim primjerima su rezultat programskih komandi i parametara prikazane su crnom bojom. Rastereti koji dočaravaju sivilo rezultat su programskih komandi, također su prikazani crnom bojom.

### I. POGLAVLJE-Programiranje grafike

Svi primjeri za Postscript se pišu u tekst editoru koji može zapisati tekst u čistoj ASCII formi, što je jako bitno za postscript. Da bi se uvidio rezultat pisanog Postscript programa može se poslati prema Postscript printeru s programom za preuzimanje nekog operativnog sustava ili se pak može vidjeti na sustavima koji podržavaju Display Postscript. Najčešći početak PostScript stranice je „moveto“ komanda. Ova komanda postavlja početak novog puta grafike u točki koja je određena parametrima pisanim ispred komande „moveto“.

Grafička stranica šalje se na ispisni uređaj: ekran, pisač, fotoosvjetljivač, već prema tome kako je ispis zamišljen da se realizira.

Linija se najčešće prikazuje kao puna, ali moguće ju je pretvoriti u crtkanu, točkastu ili kombinaciju kraćih i dužih crtica.

Bezierova krivulja osnova je mnogih zaobljenih linija kao na primjer, ovojnica slovnih znakova. Bezierova krivulja je polinom trećeg stupnja. Definirana je s četiri točke: prva, od koje točke počinje, zadnja, gdje završava krivulja te dvije tangentne točke koje određuju smjer napredovanja linije. Komanda „curveto“ koja opisuje stazu Bezierove krivulje ima samo tri para x/y točaka. Prva Bezierova točka se ne zadaje već je to ona točka koja je zadnja ostala u upotrebi.

PostScript tipografija je riješena Bezierovim krivuljama.

Grafike rađene pomoću računala prepune su likova s ponavljanim oblicima. Dizajneri najčešće planiraju ponavljanje crteža po nekom pravcu, krivudavom putu, kružnici,

simetriji. Ako nam je namjera da višestruko i precizno oblikujemo lik u pravilnom pomaku tada je računalo idealno da samo izračuna te korake.

U PostScriptu postoje četiri vrste stacka: stack parametara, stack rječnika, stack grafičkih stanja i izvršni stack. Stack parametara je spremnik (memorijski lanac) podataka i rezultata za skoro sve PostScript komande (operatore). Stack riječnika uspostavlja parove između imena varijabli i procedura sa njihovim sadržajem. Stack grafičkih stanja pamti programirani put i njegov način iscrtavanja. PostScript izvršava samo ono što je na vrhu izvršnog stacka koji mu služi kao radni stack. On je praktički transparentan za programera za razliku od ostalih.

PostScript jezik zasnovan je na stack procedurama i to najviše za stack parametara. Interpreter dodaje novi objekt (parametre, imena) iz programa i postavlja ga na vrh stacka gurajući prijašnje vrijednosti u stacku za jedno mjesto dublje. Kada PostScript interpreter pokrene izvršenje komande uzima se podatak sa vrha stacka i uklanja.

## II. POGLAVLJE-Programiranje tipografije

Slovni znakovi oblikuju se unutar "četverca". Četverac je pravokutnik unutar kojeg se postavljaju slika slovnog znaka i njegova geometrija. Tokom pisanja teksta, kada definiramo visinu slova, mi definiramo visinu četverca a ne visinu slike slova. Slika slova, za većinu slova, smještena je unutar četverca, a neki slovni znakovi djelomično ili u cijelosti (neki samostalni akcenti) izlaze iz četverca. Dno četverca leži na pismovnoj liniji, donji lijevi ugao četverca je nulta točka slovnog znaka. Debljinska vrijednost slova uglavnom uključuje cijelu širinu slike slova sa dodatnom bjelinom do slijedećeg slovnog znaka. Nulta točka pozicioniranja slijedećeg slova u tekstu je na točki debljinske vrijednosti prethodnog slova.

Latiničko pismo je četverolinijsko: pismovna linija, krov verzala, krov kurenta i descender. Visina slike verzalnih slova (A B C D...) je oko 70% četverca, visina kurantnih slova (a c e i m...) je do 50% četverca, spuštanje descendera u kurentnim slovima (j g...) je od 20 do 30%. Točne veličine za karakteristična slova i neke fontove ilustrirali smo programom na ovoj stranici. Dizajneri su dodali još četiri linije: dvije linije za krov nekih kurentnih slova (t b d f...), liniju akcenta i liniju spajanja kurentnih slova nekih rukopisnih fontova.

Da bi program radio treba prije njegovog slanja na PostScript uređaj (printer, fotojedinicu...) poslati fontove sa programom kao što je Downloader. To je jednostavnije nego pakirati fontove unutar našeg programa jer je jedan znak u fontu PostScript program za sebe. Čitljivost ovisi o pravilnom odabiru visine slova. Slova u računarskoj grafici, još od fotosloga treće generacije, određena su putanjom ovojnice na različite načine: pravci, dijelovi kružnice. PostScript koristi Bezierovu stazu. Slovni znak najčešće se prikazuje kao popunjen prostor omeđen unutarnjom i vanjskom ovojnicom.

PostScript posjeduje aritmetičke operatore (sub, add, mul, div...), stack operatore (exch, dup, pop...), relacijske operatore, logičke operatore, operatore uvjetnog izvršavanja procedura, operatore polja, string operatore i mnoge druge. Pomoću relacijskih operatora se uspoređuju se dva elementa na stacku. Rezultat te usporedbe (relacije) je logički sud koji može biti istina ili laž i pojavljuje se na stacku sa riječima true ili false. Te

logičke sudove najčešće koriste operatori uvjetnog grananja if i ifelse, kao i logički operatori not, and, or i xor.

Tekst može ući u PostScript program preko ulazne tekstualne datoteke (file) koja se prethodno napunila sa nekim tekst editorom, pisanjem rukom, programskim zapisivanjem ili generiranjem podataka.

Algoritam za dijeljenje riječi po pravilima hrvatskog pravopisa objavljen je u knjizi "Stolno izdavaštvo-DeskTop Publishing", V. Žiljak, 1989. na strani 132 i 133. Tamo je prikazan dijagram toka kao i pripadni Basic program osnovnog dijeljenja. Ovdje je taj algoritam usavršen i isprogramiran kao PostScript procedura koju kasnije mogu pozivati programi za automatsko dijeljenje riječi na kraju retka.

### **III: POGLAVLJE-Programiranje piksel grafike**

Za svaki pojedinačni piksel karakteristično je da na cijeloj svojoj površini ima jednoliku sivu boju. Vrijednost sive boje, odnosno sivoće piksela daje jednim brojem u jednom bajtu a to znači da je raspon od bijelog do crnog razdjeljen nekontinuirano na 265 stepenica sivog tona. U digitalnoj repro fotografiji koristi se termin „gray scale” odnosno siva skala kako bi se razlikovala od konturiranog tonskog prijelaza sivoće kakvu imamo u tradicionalnoj fotografskoj tehnici.

Ljudsko oko razlikuje oko 50 nijansi sive skale pa razdiobu od 256 doživljavamo kao kontinuirani ton. Slika se određuje nizom brojki koje pravilno grade ukupnu površinu slike od gornje lijevog ugla, horizontalno desno tvoreći retke od vrha slike do dna, završavajući donjim desnim uglom kao zadnjim pikselom.

Prikaz slike na ekranu ili pisaču traži podatke o broju piksela u retku, broju stupaca, veličini otisnutog piksela i položaju slike na stranici. Interpretacija slike tiskom, metodom piksela i raster, uzima se u obzir nekoliko faktora: reprodukcija detalja na slici, zauzetost memorije računala, brzina procesora računala, ograničenja tiskarskog procesa digitalnog i analognog. Slika se češće naziva original, obično se unosi u računalo skeniranjem. Pri tome se određuje veličina piksela, to jest, elementima kvadratična površina od kojih će se sastojati digitalni zapis slike. Na površini svakog elementarnog djelića slike integralno se čita zacrnjenje i taj broj pridružuje pikselu. O detaljima ili strukturi originala unutar površine piksela, nakon skeniranja, neće postojati nikakva informacija. Razlikovanje tih detalja moguće je jedino ponovnim skeniranjem, smanjivanjem elemenarne površine čitanja, što znači da se ista slika interpretira s mnogo više podataka.

Slika u piksel grafici definira se nizom podataka koji određuju zacrnjenje pojedinog piksela u nizu.

Dvodimenzionalni raspored piksela ima dva načina slaganja. Prvi način odgovara točnom (potpunom) iskorištenju podataka u stringu slike. Drugi način odnosi se na nejednak broj piksela u slici s brojem podataka u stringu slike. Smjerovi slaganja prikazani su strelicama.

Reprodukcija fotografije danas je najčešća sa 8 bitnim razinama sivog. Ljudsko oko dobro razlikuje 50 stepenica sive skale za što bi bilo dovoljno 6 bita ( $2^6 = 64$ ). Standard od 8 bita proširen je i na boju RGB, CMYK. Crno bijele slike maloga Grge prezentirane su s jednim, dva, četiri i osam bita.

U standardnim programima za piksel grafiku, kao što je npr. Photoshop, piksl na ekranu je uvijek kvadratične dimenzije. Veličina piksla se zadaje implicitno preko pojma rezolucije, odnosno brojem piksla po nekoj dužinskoj mjernoj jedinici npr. po inchu ili centimetru.

Piksel se može opisati i kao prostor slike koji je na cijeloj svojoj površini istog tona. Skaniranjem, prostor piksla integrira sve tonove na tom području, usrednjuje ih u samo jednu vrijednost. Ukoliko je piksel grub (to je relativno) tada su nakon skaniranja ili resempliranja zauvijek izgubljeni detalji manji od piksla.

Karla Žličarić