

Što je PostScript i što nam nudi?

PostScript je jezik za opis stranice koji brzo i jednostavno prikazuje grafiku, a tek u drugom koraku njegovu apstraktnu strukturu. Ne zahtjeva prethodno znanje korisnika o programiranju računala, već sklonosti prema grafičkoj umjetnosti i tipografiji. PostScript je namijenjen je samo području oblikovanja grafike na stranicama za tisak, ali je programski jezik kao i Basic, Pascal, C i ima sličnu strukturu komandi. 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). Operatori, tj. "komande" pisane su kurentnim i verzalnim slovima na engleskom jeziku (primjer: image, show, fill) ili kraticama (primjer: arc, def, div) izvedenim iz engleske riječi. Većina komandi se piše s nekoliko engleskih riječi ili kratica koje su sastavljene u kompaktni oblik (pr. setlinewidth, rlineto). Parametri, ili operandi su najčešće numeričke veličine pisane ispred komandi, a određuju način na koji komande djeluju, primjerice gdje komanda počinje i završava, koliko traje, kako se prikazuje i kako djeluje. U pisanju se kao razdjelna oznaka između parametara i komandi mogu koristiti razmak između riječi, tabulator ili kod za novi redak. Niz komandi može se kontinuirano pisati u istom retku. Preglednije pisanje programa omogućuje upotreba tabulatora i "cariage return" (cr) koda. Znak postotka % određuje početak komentara koji služi programeru kao podsjetnik, ili za olakšavanje snalaženja u programu i on nema utjecaja na PostScript komande, a komentar prestaje aktiviranjem (cr) prijelaza u novi redak. Koordinatni sustav PostScripta je zasnovan u mjernim veličinama "točka" sa X/Y koordinatama, a početak je u donjem lijevom dijelu stranice. Standardna veličina točke (tako je i u našim primjerima) određena je preko inča. Inč je podijeljen u 6 dijelova nazvanih pica (pajk - engleski cicero), a pica u 12 točaka. Tako inč ima 72 točke, odnosno jedna točka je 0.353 mm. Kod korištenja decimalne podjele primjeri imaju mrežu 100 x 100 točaka ili 35,3 x 35,3 mm. Svi primjeri se pišu u tekst editoru koji može zapisati tekst u čistoj ASCII formi, odnosno bez ikakvih tipografskih zahvata. Da bi se vidio rezultat tako pisanog PostScript programa može se poslati prema PostScript printeru sa programom za download sa bilo kojeg operativnog sustavu, ili se može vidjeti na sustavima koji podržavaju Display PostScript. Preporučuje se kao najlakše rješenje rad preko programa Ghostscript koji je PostScript Level 2 interpreter za Mac, Windows, Unix, Amiga i Atari platforme, a sve informacije o njemu mogu se naći na internetu.

Programiranje grafike

Najčešći početak PostScript stranice je *moveto* komanda. Ova komanda postavlja početak novog puta grafike u točki koja je određena parametrima pisanim ispred komande *moveto*. Pri određivanju početne točke pomažu još i komande *lineto*, *stroke* i *showpage* koje imaju svoje zadane funkcije. Pomicanje do sljedeće točke za neku udaljenost horizontalno ili vertikalno postiže se komandom *rlineto*. Ili, parametri komande *rlineto* određuju za koliko je pomak u vodoravnom ili okomitom smjeru od zadnje točke gdje smo se zaustavili. Zatvaranje lika demonstriramo komandom *closepath*. Komanda nema parametre već potraži polaznu točku u nizu prije zadnjeg pozicioniranja startne točke, na primjer točka nastala komandom *moveto*, i povuče liniju do nje. Ako debljina linije nije definirana ona je debela jednu točku kao što smo imali linije na prijašnjoj stranici. Željena debljina linije određuje se komandom *setlinewidth*. Komanda *fill* omogućuje popunjavanje i bojanje likova. Bojanje linija i zatvorenih površina, u željenim razinama sivog, postiže se komandom *setgray* koja postavlja vrijednost svjetline sivog tona. Linije se iscrtavaju komandom *stroke*. Njihova debljina se određuje komandom *setlinewidth*. Početni i završni oblik linije, što postaje važno kod debljih linija, može se oblikovati komandom *setlinecap*. Komanda ima jedan parametar koji može imati vrijednosti 0 (kvadratni rub koji završava s definicijom linije), 1 (zaobljeni rub radijusa poludebljine linije) i 2 (kvadratni završetak ali produžen za polovicu debljine linije). Zadnji oblik je važan u oblikovanju spoja vertikalnih i vodoravnih linija kakve spojeve najčešće imamo u slaganju tabličnih linija. Način spajanja linija rješava se komandom *setlinejoin*. Komanda ima jedan parametar koji može poprimiti tri vrijednosti: 0 (linije se spajaju tako da se njihova slika produžuje do tvorbe šiljastog vrha), 1 (dvije linije zatvaraju se kružnim oblikom čiji dijametar je jednak debljini linije) i 2 (vrh spoja dviju linija je okomit na simetralu kuta spajanja, a dužina je jednaka debljini linije (tupo spajanje)). Linija se najčešće prikazuje kao puna, ali moguće ju je pretvoriti u crtkanu, točkastu ili kombinaciju kraćih i dužih crtica. Izmjenično iscrtavanje crnih i bijelih crtica postiže se komandom *setdash*. Likovi: krug, kružnica, isječci kruga i njihovi dijelovi programiraju se komandama *arc*, *arcn* i *arcto*. Komande *arc* i *arcn* imaju 5 parametara: koordinate središta i radijus zadaju se u točkama, a kut početka i kut završetka luka kruga zadaju se u stupnjevima. Bezierova krivulja osnova je mnogih zaobljenih linija kao na primjer, ovojnica slovničkih 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. Ako želimo zapamtiti put i način iscrtavanja za kasniju upotrebu možemo upotrijebiti komandu *gsave* za spremanje tekućeg grafičkog stanja i komandu *grestore* koja obnavlja (restaurira) zadnje spremljeno grafičko stanje. Tako je moguće spremati više grafičkih stanja i restaurirati ih po redoslijedu spremanja. Procedure su skup rutina sa korisničkim imenom upotrebljavaju se, pozivaju, kao i bilo koja PostScript komanda. Matematičke operacije: zbrajanje, oduzimanje, množenje, dijeljenje, korjen, arkus-tanges imaju parametre slično kao i u drugim programskim jezicima. Zbrajanje uzima dva podatka s vrha stacka, zbraja ih, a rezultat stavlja na vrh stacka. Slično je i sa komandama *sub*, *mul* i *div*. Boju možemo definirati kroz različite sustave od kojih svaki ima svoju komandu za definiranje boje. Komanda *setcmykcolor* definira boju sa četiri parametra: cijan(c), magenta(m), žuta(y), crna(k). Parametri se zadaju u intervalu od 0.0. do 1.0 na inverzni način od *setgray* komanda. Kada parametar iznosi 1.0 tada je pokrivenost boje 100%. Komanda *setrgbcolor* definira boju sa tri parametra: crvena(r), zelena(g), plava(b). Komanda *sethsbcolor* definira boju sa parametrom tona (hue), zasićenosti (saturation) i svjetline (brightness).

Programiranje tipografije

Tipografija je proces kreiranja tekstovne forme, pri čemu se poštuju dobro proučene stare grafičke norme. Zadatak je tipografije da pomoću pisma na najbolji mogući način prenese informaciju. Osnovni je element tipografije slovni znak: verzal, kurent, kurziv, brojke, posebni znak, interpunkcija i razmak između riječi. 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 B C a b c.), a neki slovni znakovi djelomično ili u cjelosti (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. Neke od komandi su: *findfont* (određuje traženje fonta koji treba biti na raspolaganju programu slaganja teksta), *scalesfont* (definiranje visine četverca u točkama), *setfont* (postavljanje fonta aktivni) i *show* (prikaz fonta na postscript stranici). Slova se mogu ispisivati tipa "outline" tj. samo linijama koje leže na Bezierovim putanjama. Pri tome se mora zadati debljina linije. Naredba *charpath* služi za stvaranje ovojnice slova zadanog stringa koje će se prikazati tek upotrebom naredbe *stroke*. Sa logičkim sudom true ili false definira se vrsta outline koja će se dobiti. To ovisi o vrsti fonta koje želimo pretvoriti u outline. Postoje fontovi koji su definirani upravo kao outline sa PostScript programom koji se popune (*fill*) u momentu ispisivanja, nadalje fontovi čiji su znakovi definirani nezatvorenim linijama ili fontovi definirani bitmapom. Danas se upotrebljava samo prva vrsta fonta (outline font) za koju je rezultat naredbe *charpath* identičan i za false i true logički sud. Charpath se može kombinirati sa linijama bilo kojeg geometrijskog lika kao naprimjer kvadrat, krug, slobodna Bezierova linija. Treba voditi računa da se linija proširuje okomito na njenu putanju jednako prema centru i prema van. Popunjavanje zatvorenog puta nekim grafičkim rješenjem realizira se komandom clip bez obzira da li je osnova grafički lik ili ovojnica slova. Komanda *strokepath* omogućuje određivanje ovojnice linija, tj. stazu oko linije koja ima zadanu debljinu. Ako su linije nastale komandom *charpath* tada komanda *strokepath* daje nove površine oko linija koje su tako nastale. Ta se površina može popunjavati, bojati, podlagati (komande fill i clip). Komanda *makefont* transformira font u drugi font po transformacijskom polju (matrici). Transformacija fonta po transformacijskom polju izvodi se tako da se transformira koordinatni sustav četverca svakog znaka u fontu po navedenim formulama, a postojeći koordinatni sustav ostaje netaknut. Sa *makefont* naredbom stvaraju se novi deformirani fontovi. Ako želimo vodoravno spacionirati tekst možemo to raditi sa naredbom *ashow*. Parametri dx i dy su relativni pomaci po x odnosno y osi u odnosu na pismovnu liniju za svaki znak iz zadanog stringa. Kada je definiran samo dx parametar, a dy je 0 tada se radi o horizontalnom spacioniranju. Naredba *stringwidth* daje zbroj širine svih debljinskih vrijednosti u stringu zadanog fonta odnosno jednak je relativnom pomaku tekuće pozicije na pismovnoj liniji koji bi se dogodio nakon naredbe show. Naredba *kshow* je naprednija od svih dosadašnjih naredbi koje se tiču kontrole ispisa znakova iz stringa. Ona daje mogućnost izvršavanja programske procedure između svakog znaka stringa i to na ovaj način: 1. Ispiše se prvi znak stringa, a tekuća pozicija se pomakne sa širinu tog znaka; 2. Postavlja se na stack dekadski ASCII kod prvog znaka, a potom drugog znaka; 3. Izvršava se procedura {proc}; 4. Ispisuje se drugi znak... 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. Stvaranjem biblioteke procedura obogaćujemo i olakšavamo programiranje složenih PostScript programa. Procedura *CRODIJELI* daje na stacku riječ koja u sebi ima znak crtice na mjestima dozvoljenog dijeljenja riječi po hrvatskom pravopisu.

Programiranje piksel grafike

Slika se određuje nizom brojaka koje pravilno grade ukupnu površinu slike od gornjeg lijevog ugla, horizontalno desno tvoreći retke od vrha slike do dna, završavajući s 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. Za svaki pojedinačni piksel karakteristično je da na cijeloj svojoj površini ima jednoličnu sivoću. Vrijednost sivoće piksela dana je jednim brojem u jednom bajtu a to znači da je raspon od bijelog do crnog razdijeljen nekontinuirano na 255 stepenica sivog tona. Interpretacija slike tiskom, metodom piksela i raster, uzima u obzir nekoliko faktora: reprodukcija detalja na slici, zauzetost memorije računala, brzina procesora računala, ograničenja tiskarskog procesa digitalnog i analognog. 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. Kada je manja, popunjavanje prestaje na podatku koji zadnji popunjava zadanu matricu, a ako je veća onda image naredba počinje nanovo uzimati podatke od početka stringa dok se ne popuni zadana matrica piksela. U oba načina slaganja piksela popuniti će se dvodimenzionalni oblik slike. Reprodukcija fotografije danas je najčešća sa 8 bitnim razinama sivog. Ljusko 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. U standardnim programima za piksel grafiku, kao što je npr. Photoshop, piksel na ekranu je uvijek kvadratične dimenzije. Veličina piksela se zadaje implicitno preko pojma rezolucije, odnosno brojem piksela po nekoj dužinskoj mjernoj jedinici npr. po inču ili centimetru. Piksel se može opisati i kao prostor slike koji je na cijeloj svojoj površini istog tona. Skaniranjem, prostor piksela 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. Veličina piksela određuje se ispisom, prikazom slike. Dobri rezultati se dobe ako je ispis slike oko 300 piksela po inču. Budući da je kvaliteta otiska određena veličinom piksela, prilikom skeniranja fotografije, rezolucija skeniranja podređena je dvjema brojkama: povećanje (smanjenje) originala i broja piksela po dužnom inču (najčešće 300). U obzir može biti uzet i treći faktor - udaljenost gledanja otiska. Značajan je ako se otisci gledaju iz veće udaljenosti (plakati) jer dozvoljavaju proporcionalno povećanje dimenzije piksela u otisku.