

Što je PostScript i što nam nudi?

1. Što je PostScript?

PostScript je idealni alat za sve koji žele eksperimentirati u računarskoj tipografiji, grafici, slici, rasterima i bojama. 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 repro-fotografije.

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 odnosno komandu pisane su kurentnim i verzalnim slovima na engleskom jeziku ili kraticama izvedenim iz engleske riječi. Većina komandi sastavljena je od nekoliko engleskih riječi ili kratica, ali tako da se te riječi i kratice pišu kompaktno kao jedna tipografska riječ (npr. **setlinewidth**, **rlineto**). Parametri (ili operandi) su najčešće numeričke veličine pisane ispred komandi, a određuju način djelovanja komande (npr. gdje komanda počinje, gdje završava, koliko traje, kako se prikazuje, kako djeluje). U pisanju se jednakovrijedno koriste razmak između riječi, tabulator ili kod za novi redak, kao razdjelna oznaka između parametara i komandi. Znak postotka (%) određuje početak komentara koji služi programeru kao podsjetnik ili za olakšavanje snalaženja u programu. Taj komentar nema utjecaja na PostScript komande. Komentar prestaje aktiviranjem prelaza u novi redak.

Koordinatni sustav PostScripta zasnovan je u mjernim veličinama „točka“ sa X/Y koordinatama, a početak je u donjem lijevom dijelu stranice. Standardna veličina točke 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.

2. Programiranje grafike

Kako bismo vidjeli rezultate PostScripta, potrebno je poslati kod u PostScript printer ili imati sustav koji podržava Display PostScript.

Najčešći početak PostScript stranice je **moveto** komanda. Ta komanda predstavlja početak novog puta grafike u točki koja je određena parametrima pisanim ispred komande **moveto**. Sama komanda nije prikazala neku točku na ekranu, već je samo odredila početak nekog puta. Dužine se povlače od točke do točke koje su određene s obje koordinate: horizontalna i vertikalna. To omogućuje icrtavanje istim alatom kose, vodoravne i okomite linije. Od točke koja je posljednja određena povlači se dužina do točke koja je definirana komandom **lineto**. Premda su ove komande odredile put spojivši četiri točke, ipak nisu postale vidljive. Sama linija je bezdimenzionalna dok joj se ne pridruži debljina. Komanda **stroke** omogućuje prikazivanje linija. Sama komanda nema parametara. **Stroke** iscrtava liniju prema ranije postavljenim karakteristikama za debljinu, boju, sivilo. Ako prethodno nije ništa određeno, tada će se iscrtati crna linije debljine jedne točke. Grafička stranica šalje se na ispisni uređaj: ekran, pisač itd., već prema tome kako je ispis zamišljen da se realizira. Komanda **showpage** briše postojeće stanje i postavlja parametre za ispis sljedeće stranice.

Nakon što se odredi polazna točka, 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. Višestruko korištenje to komande kod crtanja oblika zahtjevno je zbog potrebnog znanja o potrebnoj udaljenosti od prethodne do sljedeće točke u nizu kako bi se dobio pravilan oblik. Trebali bi izračunati koliki je pomak gore ili dolje, lijevo ili desno da dođemo do početne točke. Zatvaranje lika demonstriramo komandom **closeptah**. Komanda nema parametre, već traži polaznu točku u nizu i liniju povučenu do nje. Ako debljina linije nije definirana, ona je debela jednu točku (kao i ranije). Željena debljina linije određuje se komandom **setlinewidth**, npr. od 0.5 do 5 točaka za debljinu linija kod horizontalnih linija, a kod vertikalnih od 6 do 10 točaka.

Posotje komande koje olakšavaju stvaranje određenih likova (**arc**, **arcn**, **arco**). Koriste se za prikaz kruga, kružnice i kružnih isječaka. Parametri tih komandi određuju središte, radijus, kut početka i završetka kružnog luka. Osnova mnogih zaobljenih linija je Bezierova krivulja. Bezierova krivulja polinom je trećeg stupnja. Određuje se pomoću četiri točke (druga i treća predstavljaju tangente točke). Komanda koja opisuje putanju Bezierove krivulje ima samo tri parametra jer se za prvu točku uvijek uzima zadnja točka koja se koristila.

Za ponavljanje već iscrtanih likova ili oblika često se koristi komanda **repeat**. Osim nje, korisne komande su još i **translate** i **rotate**. Te komande zapamte debljinu linija, svjetlinu i ostala svojstva lika. Komandom **scale** možemo (proporcionalno ili neproporcionalno) povećavati ili smanjivati likove. Komande **stack** ponašaju se slično kao i komande za ponavljanje, ali omogućuju izmjenu parametara prilikom kopiranja. Postoje četiri vrste **stack** komandi: **stack** parametara, rječnika, grafičkih stanja i izvršni **stack**. Boju možemo odrediti komandom **setmykcolor**, koja se služi CMYK sustavom.

3. Programiranje tipografije

Tipografija je proces kreiranja tiskovne forme čiji je zadatak da pomoću pisma na najbolji mogući način prenese informaciju. Osnovni element tipografije jest slovni znak (verzali, kurenti, kurzivi...). Slovni znakovi oblikuju se unutar četverca. Četverac je pravokutnik unutar kojeg se postavljaju slika slovnog znaka i njegova geometrija. Tijekom pisanja teksta, kada definiramo visinu slova, mi definiramo visinu četverca, a ne visinu slike slova. Slika slova (za većinu) smještena je unutar četverca (A B C a b c...), a neki slovni znakovi djelomično ili u sjelosti 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 sljedećeg slovnog znaka. Nulta točka pozicioniranja sljedećeg slova u tekstu je na točki debljinske vrijednosti prethodnog slova.

Latinično pismo je četverolinijsko: pismovna linija, krov verzala, krov kurenta i descender. Visina slike verzalnih slova (A B C D...) je oko 70% četverca. Visina kurentnih slova (a c e i m...) je do 50% četverca, a spuštanje descendera u kurentnim slovima (j g...) je od 20% do 30% četverca. Za krov nekih kurentnih slova (t b d f...) dodaju se još dvije linije, a za neke neke (Ž Č...) se dodaju linije akcenta.

Komanda **findfont** određuje traženje fonta koji treba biti na raspolaganju programu slaganja teksta. Komanda **scalefont** služi za definiranje visine četverca u točkama, a komanda **setfont** služi za postavljanje nekog fonta aktivnim, dok komanda **show** služi za prikaz fonta na PostScript stranici.

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. Za to je dovoljna komanda **show**. Slova se mogu ispisivati tipa „outline“, tj. samo na linijama koje leže na Bezierovim putanjama, pri čemu se mora znati debljina linije. Naredbom **charpath** stvaraju se ovojnice slova zadanog stringa koje će se prikazati tek upotrebom naredbe **stroke**. Sa logičkim sudom *true* ili *false* definira se vrsta *outlinea* koja će se dobiti. To ovisi o vrsti fonta koji želimo pretvoriti u *outline*. Postoje fontovi koji su definirani upravo kao *outline* sa PostScript programom koji se popune (*fill*) u trenutku ispisivanja. Danas se upotrebljava samo prva vrsta fonta (*outline font*) za koju je rezultat naredbe **charpath** identičan i za *true* i za *false* logički sud.

Slovni znak najčešće se prikazuje kao popunjen prostor omeđen unutarnjom i vanjskom ovojnicom. tekst može ući u PostScript program preko ulazne tekstualne datoteke koja se prethodno napunila s nekim tekst editorom, pisanjem rukom, programskim zapisivanjem ili generiranjem podataka.

4. Programiranje piksel grafike

U ovom dijelu razmatra se slika sastavljena od sivih kvadratića nazvanih *piksel* (*picture element*). Slika u piksel grafici definira se nizom podataka koji određuju zacrnjenje pojedinog piksela u nizu. 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 je nekontinuirano na 265 stepenica sivog tona. U digitalnoj reprofotografiji koristi se termin „siva skala“ kako bi se razlikovao od kontinuiranog tonskog prijelaza sivoće kakvu imamo u tradicionalnoj fotografskoj tehnici. Ljudsko oko razlikuje oko 50 nijansi siva skale, pa razdiobu od 256 doživljavamo kao konitnuirani ton. Slika se određuje nizom brojki 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. Interpretacija slike tiskom, metodom piksela i rastera, uzima u obzir nekoliko faktora: reprodukciju detalja na slici, zauzetost memorije računala, brzinu procesora računala, ograničenja tiskarskog procesa digitalnog i analognog. Slika, češće zvana original, obično se unosi u računalo skeniranjem. Pri tome se određuje veličina piksela, tj. elementarna kvadratična površina od kojih će se sastojati digitalni zapis slike. Na površini svakog elementarnog djelića slike integralno se čita zacrnjenje i taj broj pridruži pikselu. O detaljima ili strukturi originala unutar površine piksela, nakon skeniranja, ne će postojati nikakva informacija. Razlikovanje tih detalja moguće je jedino ponovnim skeniranjem ,smanjivanjem elementarne površine čitanja, a to znači da se ista slika interpretira s mnogo više podataka.

Uvriježeno je razmišljanje da je piksel kvadratičnog oblika jer mnogi programi imaju samo takvu mogućnost manipulacije njime. Pikseli su definirani kao paralelogrami, pa svaki pojedini piksel može poprimiti zakrenute deformirane oblike.

U standardnim programima za piksel grafiku, kao što je npr. Photoshop, piksel na ekranu je uvijek kvadratične dimenzije. Veličina piksela zadaje se implicitno preko pojma rezolucije, odnosno brojem piksela 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 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 zauvijek izgubljeni detalji manji od piksela. Četiri 8-bitne slike demonstriraju promjene detalja smanjenjem memorije, odnosno smanjenjem broja piksela iste slike za faktor 10 od 1 MB do 1 kB.

Veličina piksela određuje se ispisom, prikazom slike. Dobri rezultati dobiju se ako je ispis slike oko 300 piksela po inču. I budući da je kvaliteta otiska određena veličinom piksela, prilikom skaniranja fotografije, rezolucija skiniranja podređena je dvjema brojkama: povećanja (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.