

1 Sta je ono sto izdvaja Perl?



Figure 1: Logo CPAN skladišta

"Velika Perl Arhiva" (eng. Comprehensive Perl Archive Network(CPAN)) predstavlja online skladište Perl programa, dokumentacije i modula koji olakšavaju implementaciju i održavanje Perl projekata. Arhiva čuva oko 180 hiljada modula otvorenog koda napisanih od preko 12 hiljada programera.

Glavni zadatak CPAN-a je da omogući programerima da pronadju module koji nisu uključeni u paket osnovnih modula. Paket osnovnih modula se dobija instalacijom Perl interpretatora. Neki od njih su

Modul pod imenom CPAN.pm omogućava lako preuzimanje i instaliranje drugih Perl modula.

1.1 Parsiranje tekstualnog sadržaja

Prve verzije Perl programskog jezika su u fokus stavljale obradu i manipulaciju tekstualnog sadržaja. Sama skracenica PERL (Practical Extraction and Reporting Language) prikazuje osnovne namene jezika, a to su izvlačenje nekog sadržaja i generisanje izveštaja na osnovu tog sadržaja. Zbog svoje čiste i jednostavne sintakse, neki autori smatraju da je Perl nezamenljiv izbor kada je u pitanju parsiranje fajlova.

Naredni primer prikazuje kako na jednostavan način možemo izračunati broj pojavljivanja svake reči u tekstu, pri čemu se za reč smatra niska karaktera koja se sastoji samo od malih i velikih slova abecede.

Listing 1: Brojanje reci u tekstualnom fajlu

```
# ...
while (my $line = <$fh>) {
    chomp $line;
    foreach my $str (split /\^[A-Za-z]+$/, $line) {
        $count{$str}++;
    }
}
# ...
```

Dakle, prvo prolazimo kroz fajl (predstavlja referencu na fajl koji smo prethodno otvorili) liniju po liniju, zatim iz linije uklanjamo znak za novi red, a potom za svaku reč (smeštenu u promenljivoj \$str) koja odgovara navedenom regularnom izrazu, ažuriramo strukturu u kojoj čuvamo reč i broj izračunatih pojavljivanja. Rezultat rada navedenog dela koda je smešten u promenljivoj \$count.

1.2 Jednostavno korišćenje regularnih izraza

Važnu ulogu pri manipulaciji tekstualnog sadržaja igraju regularni izrazi. Regularni izrazi su šabloni koji mogu biti prepoznati u nizovima karaktera. Rezultat rada operacije prepoznavanja šablona jeste tačno ili netačno, odnosno da li niz karaktera odgovara datom ablonu ili ne. Perl ima ugrađene mehanizme koje čine rad sa regularnim izrazima lakšim, nalik na alate `grep`, `sed` i `awk`.

Osnovni operatori primene regularnih izraza, koje Perl nudi, jesu `=`, `!` i `%`. U izrazu `$str =~ m/regex/` se proverava da li se dati string `$str` podudara sa regularnim izrazom `regex`. Regularni izrazi se navode između dve kose crte. U slučaju da odgovara izraz vraća 1, a nula inače. Operator `!` radi suprotno, u slučaju podudaranja vraća 0, a 1 inače.

Postoje tri osnovne operacije sa regularnim izrazima:

- Provera podudaranja - `m/regex/`
- Zamena reči - `s/regex/word/`
- Translacija - `tr/regex/word/`

Zamena reči funkcioniše tako što se sva podudaranja regularnog izraza u stringu zamene navedenim stringom. Na primer u izrazu `"I like cats!" =~ s/cats/dogs`, deo stringa `cats` će biti zamenjen sa `dogs`, i novodobijeni string predstavlja rezultat izraza.

Translacija je slična zameni reči, sa tim što ne koristi regularne izraze pri likom pretrage stringa.