STUDENT: **Zoran Novaković**
JMBAG: **0036390963**
SMJER: **Industrijska elektronika**
PREDMET: **Sustavi za praćenje i vođenje procesa**

SEMINARSKI RAD

# KODNI SUSTAVI ZA PRIKAZ ZNAKOVA

Zagreb, 5. lipnja, 2005.

SADRŽAJ

# 1. UVOD

Kodiranje podrazumijeva proces prikaza informacija u nekom obliku. Jezik ljudi je kodni sustav kojim predstavljamo informacije kao nizove leksičkih jedinica, a njih pak kao zvukove odnosno mimiku. Pisani jezik je izvedeni sustav kodiranja kojim te nizove leksičkih jedinica, zvukova i gesti predstavljamo grafičkim simbolima koji sačinjavaju neki sustav pisma.

U računalnim sustavima, kodiramo pismo na način da predstavljamo grafeme i druge elemente pisanog teksta kao nizove znakova, jedinica tekstualne informacije unutar nekog sustava za prikaz teksta. Ovi znakovi su u računalu prikazani na jedini način s kojim računalni sistemi znaju baratati: binarnim brojevima. Kodni sustavi za prikaz znakova rade upravo to.

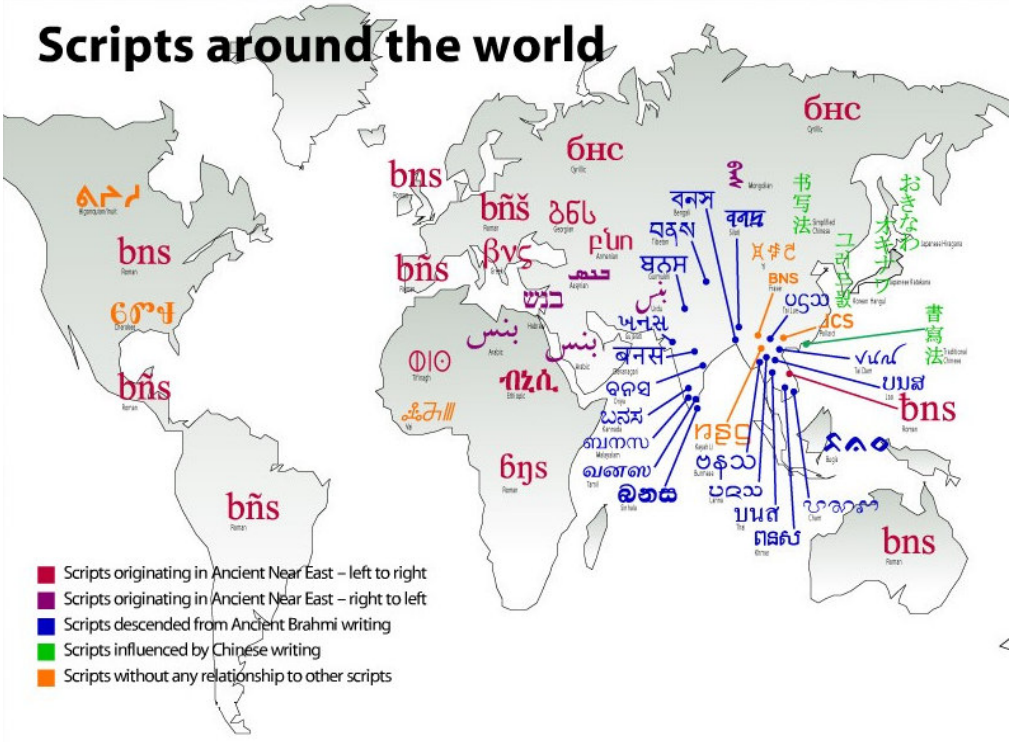
Svako kodiranje uključuje dvije osnovne komponente: niz znakova i nekakav sustav za njihovo prikazivanje u obliku obradivih jedinica unutar računala. Naravno, ne postoji predodređen način na koji se ovo radi. ASCII standard je jedan od sustava, ali ne i jedini. Na taj način, isti niz brojeva može se protumačiti na beskonačno mnogo načina, zavisno o konvencijama koje se pretpostavljaju. U najjednostavnijem slučaju, koji je još uvijek najraširenij, jedan oktet bitova (bajt) odgovara jednom znaku prema nekoj kodnoj tablici. Ovako je moguće prikazati 256 različitih znakova. Za pravilan način dekodiranja, mora biti poznato koja tablica se koristi. Za HTML dokumente, takvu informaciju bi morao slati server uz sam dokument u tzv. HTTP zaglavljima.

## 2. INDUSTRIJSKI STANDARDI I STANDARDI OPĆENITO

Dok je ranije ASCII kodiranje bilo najučestalije, danas je raširen ISO Latin 1 način kodiranja, koji se može smatrati ekstenzijom ASCIIa. ASCII je bio minimalno dostatan za prikaz teksta na američkom engleskom. Nije bio dostatan za britanski engleski, a kamoli za izdavaštvo na engleskom govornom području ili gotovo bilo kojem drugom jeziku. Ubrzo su prodrli novi standardi i to iz dva izvora: tijela za standardizaciju i nezavisni proizvođači softvera i hardvera. Tako su IBM (codepage 437, 852, 855), Apple i Microsoft (codepage 1252 = Western = Latin 1) stvarali standarde usput kako su im zatrebali za proizvode koje su izdavali. Ako bi neki takav standard postao široko prihvaćen, nacionalno standardizacijsko tijelo ga je moglo proglasiti nacionalnim standardom, a takvi su standardi ponekad postajali i međunarodni. Uz cijelo mnoštvo ovako nastalih standarda, događalo se da neki standardi podržavaju određene znakove koji ne postoje u drugih. Također se dešavalo da su podržani isti setovi znakova, ali na nekompatibilan način. Ovi problemi se danas rješavaju Unicode standardom, ali naslijeđeni standardi se ne mogu jednostavno zanemariti.

Svaki komercijalni softverski proizvod je eksplicitno dizajniran da podržava određeni set standarda za kodiranje znakova. Sve njegove operacije s tekstom bit će izvedene pod pretpostavkom da je aktivan jedan od tih standarda. Ako dakle podaci nisu točno ili uopće označeni kojim standardom su kodirani, doći će do neočekivanih rezultata.

Problem sa softverom baziranom na standardima je da ako je potrebno raditi sa kompletom znakova koji softver ne prepoznaje, onda ste zapeli. Ovo je posebno bio problem za lingviste koji rade sa manje zastupljenim jezicima. Takvi onda kreiraju vlastita rješenja za svoje potrebe koja su sasvim dobra za vlastitu primjenu, ali zakazuju kada dođe do potrebe za razmjenom na višem nivou. Način za izbjegavanje ovih problema je uporaba standardnog kodiranja koje uključuje sve znakove. Upravo takav tip rješenja će omogućiti Unicode koji se razvija s ciljem da ima univerzalni set znakova koji bi pokrivao potrebe na svjetskoj razini. (Više o Unicodeu kasnije.)



# 3. MODEL KODIRANJA SETA ZNAKOVA

Ranije je rečeno da kodiranje seta znakova uključuje bar dvije razine, set znakova i sustav za njihov kodirani prikaz u računalu. Ipak je potreban složeniji model koji uključuje četiri različite razine prikaza:

- repertoar apstraktnih znakova (ACR)
- set kodiranih znakova (CCS)
- obrazac za kodiranje znakova (CEF)
- shema kodiranja znakova (CES)

Repertoar apstraktnih znakova (ACR) je jednostavno skup svih znakova koje je potrebno kodirati, a koji nisu poredani nikakvim posebnim redoslijedom. On može biti zatvoren (ne postoji mogućnost dodavanja novih znakova) ili otvoren (mada se dodatni znakovi ne moraju moći dodavati u količinama po volji). Npr. Unicode ima otvoreni repertoar koji se redovito dograđuje ne bi li standard bio još univerzalniji. Repertoar znakova se može sastojati od znakova koji izgledaju jednako u nekim reprezentacijama, ali su logički različiti. Npr. veliko A u Latinici, u Ćirilici i Grčkom pismu.

Set kodiranih znakova (CCS) je samo mapiranje iz nekog repertoara u set jedinstvenih numeričkih oznaka – tipično cjelih brojeva. Ovo se često izvodi tablično i pridjeljuje jedinstven brojevni kod – „kodnu poziciju“ svakom znaku iz repertoara. Ovakav niz pozitivnih cjelih brojeva ne mora ići slijedno, štoviše većina setova ima „praznine“ kao pozicije rezervirane za kontrolne funkcije ili buduću upotrebu.

Obrazac za kodiranje znakova (CEF) je metoda (algoritam) za prikaz znakova u digitalnom obliku mapiranjem slijeda kodnih pozicija u slijed vrijednosti fiksnog podatkovnog tipa. Ove vrijednosti su znane kao kodne jedinice koje praktički uglavnom iznose 8 bita, a postoje i one od 16 tj. 32 bita. Obrazac za kodiranje može biti bez stanja, što znači da koji god znak se kodira ili slijed koji koristi su dostupni u svako vrijeme. Neki su pak modalni, što znači da se u određenom stanju mogu kodirati samo neki znakovi. Obrazac za kodiranje u ovom slučaju specificira određeni kod, čija pojava uzrokuje promjenu stanja i drugačije tumačenje sljedećih podataka.

Shema kodiranja znakova (CES) nam faktički govori kojim redom su poslagani bajtovi u 16 i 32-bitnim kodovima. Naime tijekom vremena su „bajtovi od 8 bitova“ postali značajna mjera u računalnim sistemima. Kada dakle višebajtni kod uleti u jedan takav sustav tumačenja podataka, postoje dva načina na koji on može biti tumačen. Little-endian znači da je bajt nižeg redoslijeda prvi, a big-endian obratno. CES razrješava ovu dvojbu. Unicode je jedan od rijetkih obrazaca kodiranja koji ima višebajtnu strukturu, a on podržava oba načina zapisa.

# 4. PRIMJERI ZNAKOVNIH KODOVA

## 4.1. ASCII ALIAS ISO 646

American Standard Code for Information Interchange je oznaka za repertoar znakova, set kodiranih znakova i obrazac za kodiranje. Većina postojećih kodova sadrže ASCII kao svoj podskup u nekom smislu. Naziv ACSII se koristi naširoko i često pa se tako ponekad misli na tekst općenito, ili tekstualnu datoteku za razliku od binarne.

ASCII po definiciji sadrži i set kontrolnih kodova, ali pravi repertoar znakova koji se sastoji od ispisivih znakove je sljedeći:

```
  ! " # $ % & ' ( ) * + , - . /
0 1 2 3 4 5 6 7 8 9 : ; < = > ?
@ A B C D E F G H I J K L M N O
P Q R S T U V W X Y Z [ \ ] ^ _
` a b c d e f g h i j k l m n o
p q r s t u v w x y z { | } ~
```

Set kodiranih znakova definiran ASCII standardom je sljedeći: kodne vrijednosti su dodijeljene znakovima slijedno redom koji je naveden gore, počinjući od 32 za prazninu i završavajući sa 126 (tilda). Mjesta 0 – 31 i 127 su rezervirana za kontrolne kodove koji imaju standardizirana imena i opise, ali im primjena varira.

| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0 | NUL | SOH | STX | ETX | EOT | ENQ | ACK | BEL | BS | HT | LF | VT | FF | CR | SO | SI |
| 1 | DLE | DC1 | DC2 | DC3 | DC4 | NAK | SYN | ETB | CAN | EM | SUB | ESC | FS | GS | RS | US |
| 2 | SPC | ! | " | # | $ | % | & | ' | ( | ) | * | + | , | - | . | / |
| 3 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | : | ; | < | = | > | ? |
| 4 | @ | A | B | C | D | E | F | G | H | I | J | K | L | M | N | O |
| 5 | P | Q | R | S | T | U | V | W | X | Y | Z | [ | \ | ] | ^ | _ |
| 6 | ` | a | b | c | d | e | f | g | h | i | j | k | l | m | n | o |
| 7 | p | q | r | s | t | u | v | w | x | y | z | { | \| | } | ~ | DEL |

Obrazac za kodiranje je vrlo jednostavan i prilično očit za sve kodove znakova gdje kodni brojevi ne prelaze 255: svaki kodni broj je predstavljen kao oktet (bita) iste vrijednosti. Okteti 128-255 se ne koriste u ASCIIu.

Postoji nekoliko nacionalnih varijanti ASCIIa u kojima su neki specijalni znakovi zamijenjeni nacionalnim slovima (i simbolima). Originalni ASCII je onaj pod ANSI standardom ANSI X3.4-1986. Tako se u razmjeni podataka preporučuje ne koristiti sve dostupne znakove za imenovanje datoteka i slično, jer nisu „sigurni“. Podskup „sigurnih“ znakova u ASCIIu je

```
! " % & ' ( ) * + , - . / : ; < = > ?
```

Ponekad se krivo koristi termin „8-bitni ASCII“, što je pogrešno. Pod tim se obično misli na različite znakovne kodove koji su ekstenzija ASCIIa. Repertoar znakova sadržava ASCII repertoar kao podskup i kodni brojevi za te znakove su isti kao u ASCIIu.

## 4.2. ISO LATIN 1 ALIAS ISO 8859-1

ISO 8859-1 standard (koji je dio ISO 8859 porodice standarda) definira repertoar znakova koji se naziva „Latin alphabet No. 1“, obično zvan „ISO Latin 1“, te set kodiranih znakova. Repertoar sadrži ASCII kao podskup i kodni brojevi za te znakove su isti kao u ASCIIu. Standard specificira i obrazac za kodiranje koji je sličan onome u ASCIIu: svaki znak je predstavljen oktetom.

Kao dodatak ASCII znakovima, ISO Latin 1 sadrži različite znakove s naglascima i druga slova potrebna za pisma zapadne Europe i neke posebne znakove. Ovi znakovi zauzimaju mjesta 160 – 255 i to su:

| | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | ¡ | ¢ | £ | ¤ | ¥ | ¦ | § | ¨ | © | ª | « | ¬ | ® | ¯ |
| ° | ± | ² | ³ | ´ | µ | ¶ | · | ¸ | ¹ | º | » | ¼ | ½ | ¾ | ¿ |
| À | Á | Â | Ã | Ä | Å | Æ | Ç | È | É | Ê | Ë | Ì | Í | Î | Ï |
| Ð | Ñ | Ò | Ó | Ô | Õ | Ö | × | Ø | Ù | Ú | Û | Ü | Ý | Þ | ß |
| à | á | â | ã | ä | å | æ | ç | è | é | ê | ë | ì | í | î | ï |
| ð | ñ | ò | ó | ô | õ | ö | ÷ | ø | ù | ú | û | ü | ý | þ | ÿ |

## 4.3. WINDOWS SET ZNAKOVA

Tijekom razvoja Windowsa Američki Nacionalni Institut za Standarde (ANSI) je razvijao standard koji je postao ISO 8859-1 „Latin“. Microsoft je kreirao svoju kodnu stranicu 1252 za zapadno europske jezike na osnovi ranog nacrta ANSI prijedloga i to nazvao „ANSI kodna stranica“. Kodna stranica 1252 je finalizirana prije nego ISO 8859-1 i to dvoje nisu isti: 1252 je širi od ISO 8859-1. Kasnije je Microsoft počeo koristiti „ANSI“ u smislu bilo koje Windows kodne stranice, za razliku od Unicode.

U ISO 8859-1 standardu, kodne pozicije 128 – 159 su eksplicitno rezervirane za kontrolne svrhe tj. odgovaraju kombinacijama bitova koje ne predstavljaju grafičke znakove. Takozvani Windows set znakova (WinLatin1 ili Windows kodna stranica 1252) koristi neke od tih pozicija za ispisive znakove. Windows set znakova se često naziva „ANSI set znakova“, ali ovo je pogrešno jer nije odobren od ANSIa. Korištenje kodnih znakova pod brojevima 128 – 159 u programu koji očekuje pravi ISO 8859-1 može uzrokovati različita ponašanja. Može se desiti da se ti znakovi ignoriraju, da izgledaju kao nešto što ima smisla ili da se interpretiraju kao kontrolni znakovi.


Windows

## 4.4. ISO 8859 PORODICA

Postoji nekoliko kodova znakova koji su ekstenzija ASCIIa na isti način kao što su to ISO 8859-1 i Windows set znakova. Sam ISO 8859-1 je dio veće obitelji znakovnih kodova ISO 8859 čiji kodovi proširuju repertoar ASCIIa na različite načine sa različitim posebnim znakovima (koji se koriste u raznim jezicima i kulturama). Baš kao što ISO 8859-1 sadrži ASCII znakove i kolekciju znakova potrebnih za jezike zapadne (i sjeverne) Europe, tu je i ISO 8859-2 alias ISO Latin 2 koji je konstruiran slično za jezike središnje/istočne Europe itd. ISO 8859 kodovi su izomorfni u sljedećem smislu:

- kodni brojevi 0 – 127 sadrže iste znakove kao i ASCII
- mjesta 128 – 159 su neiskorištena (rezervirana za kontrolne znakove)
- mjesta 160 – 255 su promjenjivi dio koji se koristi različito u različitim izdanjima

Iako se ISO 8859-1 koristio kao de facto predodređen za kodiranje u mnogim primjenama, on u principu nema neku posebnu ulogu. Očekivalo se da će ISO 8859-15 alias ISO Latin 9 zamijeniti ISO 8859-1, budući sadrži politički važan simbol za euro €, no čini se da ipak nema veliku praktičnu primjenu.

Sljedeća tablica daje pregled ISO 8859 alfabeta. U dokumentu „Coverage of European languages by ISO Latin alphabets“ može se vidjeti koji (ako ikoji) od alfabeta je odgovarajući za dokument na danom jeziku.

| **Dijelovi ISO 8859** | | |
|---|---|---|
| **standard** | **ime alfabeta** | **karakterističan za** |
| ISO 8859-1 | Latin alphabet No. 1 | zapadni europski |
| ISO 8859-2 | Latin alphabet No. 2 | središnji europski, istočni europski |
| ISO 8859-3 | Latin alphabet No. 3 | južni europski, malteški i Esperanto |
| ISO 8859-4 | Latin alphabet No. 4 | sjeverni europski |
| ISO 8859-5 | Latin/Cyrillic alphabet | za slavenske jezike |
| ISO 8859-6 | Latin/Arabic alphabet | za arapski jezik |
| ISO 8859-7 | Latin/Greek alphabet | za moderni grčki |
| ISO 8859-8 | Latin/Hebrew alphabet | za hebrejski i jidiš |
| ISO 8859-9 | Latin alphabet No. 5 | turski |
| ISO 8859-10 | Latin alphabet No. 6 | nordijski (sámi, inuitski, islandski) |
| ISO 8859-11 | Latin/Thai alphabet | za thai jezik |
| (dio 12 nije definiran) | | |
| ISO 8859-13 | Latin alphabet No. 7 | baltički rub |
| ISO 8859-14 | Latin alphabet No. 8 | keltski |
| ISO 8859-15 | Latin alphabet No. 9 | "euro" |
| ISO 8859-16 | Latin alphabet No. 10 | za niz jezika – albanski, hrvatski, engleski, finski, francuski, njemački, mađarski, irski, talijanski, latinski, poljski, rumunjski, slovenski |

## 4.5. DRUGE „EKSTENZIJE ASCIIa“

Uz dodatak gore opisanim kodovima postoje i druge ekstenzije ASCIIa koje iskorištavaju kodni raspon 0 – 255. Takvi su:

**DOS znakovni kodovi ili kodne stranice**

Prvotna američka kodna stranica (CP) je bila CP 437, koja ima npr. neka grčka slova, matematičke simbole i znakove koji se mogu koristiti kao elementi u jednostavnim pseudo-grafičkim oblikovanjima. Kasnije je CP 850 postao popularan, budući sadrži slova za zapadno europske jezike – uvelike ista slova kao ISO 8859-1, ali na različitim kodnim mjestima. DOS kodne stranice su prilično različite od Windows znakovnih kodova, mada se potonje ponekad nazivaju npr. CP-1252 (=windows-1252). Radi još veće pomutnje, Microsoft sada preferira naziv „OEM kodna stranica“ za DOS set znakova korišten u pojedinoj zemlji.

**MACINTOSH znakovni kodovi**

Na Macovima, znakovni kod je više jednoobrazan nego na PCima (iako postoje neke nacionalne varijante). Mac repertoar znakova je miješana kombinacija ASCIIa, slova s naglascima, matematičkih simbola i drugih sastojaka.

Općenito govoreći, potpune konverzije između navedenih znakovnih kodova nisu moguće. Neka slova koja postoje u Macintosh kodu uopće ne postoje u Latin 1 kodu. Obični tekst se može konvertirati (jednostavnim programom koji koristi konverzijsku tablicu) iz Macintosh koda u ISO 8859-1 ako takav tekst uistinu sadrži samo one znakove koji postoje u ISO Latin 1 repertoaru. Isto vrijedi za konverzije između drugih kodova.

**Standardi ISO 2022 i ISO 4873** definiraju općeniti okvir za 8-bitne i 7-bitne kodove i prijelaz između njih. Praktično je bilo napraviti donje polovice kodova jednake, a jedna od osnovnih odrednica je da su kodni brojevi 128 – 159 rezervirani za kontrolne kodove. Windows kodne stranice ne uvažavaju ovo načelo. Jedan od takvih zanimljivih kodova je i **EBCDIC** kod, definiran od IBMa i nekoć u širokoj upotrebi na mainframe računalima. EBCDIC sadrži sve ASCII znakove, ali na prilično različitim kodnim mjestima. Npr. zanimljiv je detalj da se normalna slova A – Z uopće ne pojavljuju na slijednim pozicijama. EBCIDIC postoji u različitim nacionalnim varijantama.

## 4.6. ISO 10646, UCS I UNICODE

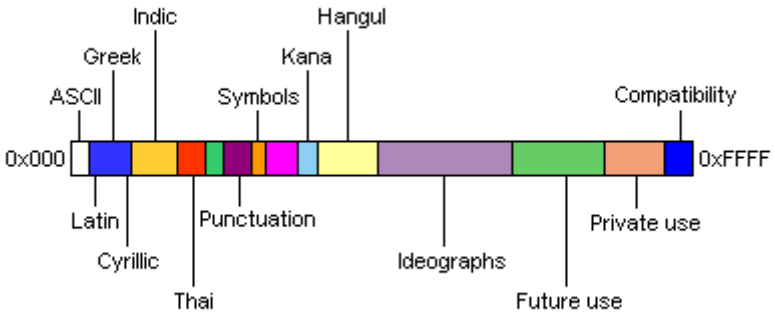
**ISO 10646** (službeno: ISO/IEC 10646) je međunarodni standard i definira univerzalni set znakova (UCS) koji je vrlo velik i rastući repertoar znakova i pripadajućih znakovnih kodova. Definirano je na desetke tisuća znakova i novi amandmani se definiraju prilično često. Među ostalima, on sadrži sve znakove iz repertoara znakova gore spominjanih kodova.

**Unicode** je standard, od strane Unicode konzorcija, koji definira repertoar znakova i set kodiranih znakova koji su u potpunosti kompatibilni s ISO 10646 i obrazac za kodiranje znakova. ISO 10646 je općenitiji i po prirodi apstraktniji, dok Unicode „nameće dodatna ograničenja na implementaciju da bi osigurao uniforman tretman znakova na svim platformama, programima i jezicima“.

Unicode rješava sve one probleme oko jednoznačnosti, transformacija među različitim kodnim tablicama, raspoloživog opsega znakova koji su isticani u dosadašnjem tekstu. Unicode standard prihvaćen je od vodećih industrijskih tvrtki kao što su: Apple, HP, IBM, Microsoft, Oracle, SAP, Sun, Sybase, Unisys i mnogih drugih. Unicode je potreban kod modernih standarda kao što su: XML, Java, ECMAScript (JavaScript), LDAP, CORBA 3.0, WML, itd. Podržan je od strane mnogih operativnih sustava i svih suvremenih pretraživača. Ugrađivanje Unicodea u klijentsko-poslužiteljske ili višeslojne (multi-tiered) aplikacije i site-ove donosi značajnu novčanu uštedu u odnosu na korištenje starih znakovnih sustava.

**Unicode konzorcij** je neprofitna organizacija osnovana u cilju razvoja, širenja i promoviranja korištenja Unicode standarda. Članstvo u konzorciju predstavlja širok spektar korporacija i organizacija u računalnoj i informatičkoj industriji, a konzorcij se financijski izdržava isključivo iz članarine. Članstvo u Unicode konzorciju je otvoreno za organizacije i individualce bilo gdje u svijetu ukoliko podržavaju Unicode standard i žele pomoći u njegovom širenju i implementaciji.

Unicode je prvotno zamišljen kao 16-bitni kod, ali proširen je tako da su trenutne kodne pozicije predstavljene cijelim brojevima u heksadecimalnom rasponu 0 – 10FFFF (decimalno 0 – 1 114 111). Taj prostor podijeljen je u 16-bitne „ravnine“. Do nedavno, upotreba Unicodea je uglavnom bila ograničena na „osnovnu višejezičnu ravninu“ (BMP) koja se sastoji od raspona 0 – FFFF.



U praksi ljudi govore o Unicodeu radije nego o ISO 10646, djelomično jer preferiramo imena radije nego brojeve, dijelom jer je Unicode eksplicitniji u značenju znakova, a dijelom jer su detaljne informacije o Unicodeu dostupne na Internetu (dok za ISO 10646 to nije slučaj – može se dobiti u ispisanom obliku od članova tijela ISO).

Prije proširenja koda iznad 16 bita, prvotno Unicode kodiranje je bilo UCS-2, koje predstavlja svaki kodni broj kad dva okteta m i n, tako da broj iznosi 256m+n. Odnosno, kodni broj je predstavljen kao dvo-bajtni cijeli broj. Termin UCS-2 bi trebalo izbjegavati jer se povezuje sa 16-bitnim ograničenjima.

**UTF-32** kodira svaki kodni broj kao 32-bitni cijeli broj, tj. kao 4 bajta. Ovo je vrlo očito i jednostavno kodiranje. S druge strane, neučinkovito je s obzirom na broj korištenih bajtova. Ako imamo normalni engleski tekst ili neki drugi tekst koji sadrži samo znakove iz tablice ISO Latin 1, dužina kodiranog niza bajtova 4 je puta duža u Unicodeu nego u ISO 8859-1 kodu. UTF-32 se rijetko koristi, osim možda u internim operacijama.

**UTF-16** predstavlja svaki kodni broj u osnovnoj višejezičnoj ravnini kao dva bajta.

Unicode se može kodirati na druge načine, kao što su sljedeći:

**UTF-8**

Kodovi znakova manji od 128 (efektivni ASCII repertoar) se predstavljaju „kao takvi“, koristeći jedan oktet za svaki kod (znak). Svi drugi kodovi su prikazani prema relativno kompliciranoj metodi, tako da je jedan znak predstavljen kao niz od dva do četiri okteta, od kojih je svaki u rasponu 128 – 255. Ovo znači, da u nizu bajtova, oni koji su u rasponu 0 – 127 (bajtovi s MSBom u nuli) izravno predstavljaju ASCII znakove, dok bajtove u rasponu 128 – 255 (bajtovi s MSBom u jedinici) treba interpretirati kao stvarno kodirane prezentacije znakova.

**UTF-7**

Svaki kod znaka je predstavljen nizom jednog ili više okteta u rasponu 0 – 127. Većina ASCII znakova je tako predstavljena, svaki po jedan oktet, ali očito je da neke vrijednosti moraju biti rezervirane za upotrebu kao „okteti za bijeg“, koji govore da oktet zajedno s određenim brojem okteta koji slijede formira više-oktetnu prezentaciju jednog znaka.

UTF-7 se vrlo malo koristi, dok se UTF-8 favorizira zbog efikasnosti.

Implementacija podrške za Unicode je dugotrajan i uglavnom postupan proces. Unicode može biti podržan od strane svih programa na svim operacijskim sustavima, iako neki sustavi dopuštaju lakšu implementaciju nego drugi. Ovo uglavnom ovisi o tome koristi li sustav Unicode interno tako da je podrška za njega već „ugrađena“.

Čak i kad je Unicode principijelno podržan, podrška uglavnom ne pokriva sve Unicode znakove. Npr. font na raspolaganju može pokrivati samo dio Unicodea koji je praktički važan u nekom području. Iz takvih razloga razni podskupovi repertoara znakova Unicodea su definirani. Na primjer, **Minimalni Europski Podskup** specificiran sa ENV 1973:1995 je trebao predstavljati prvi korak prema implementaciji velikih znakovnih setova u Europi. Nadomješten je sa tri višejezična europska podskupa (MES-1, MES-2, MES-3), definiranim u CWA 13873.

Znakovi Unicodea se ponekad referiraju prema notaciji oblika **U+nnnn** gdje je nnnn 4-znamenkasta heksadecimalna notacija kodne vrijednosti. Npr. U+0020 predstavlja znak razmaka (20H =32D). Takve notacije identificiraju znak prema kodnoj vrijednosti unutar Unicodea, bez reference na određeno kodiranje.

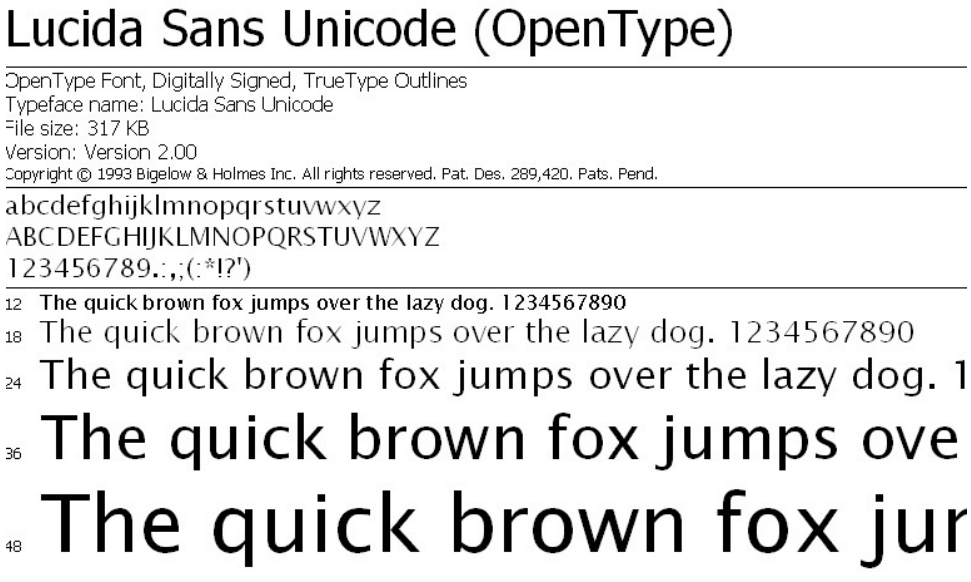
## 5. KONCEPT ZNAKA – GRAFEM, FONT

**GRAFEM – Vizualni prikaz**

Važno je razlikovati koncept znaka od koncepta grafema (glyph). Grafem je grafička reprezentacija konkretnog oblika koji znak može imati kada se ispiše ili prikaže. Na primjer, znak Z se može prikazati kao podebljani **Z** ili nakošeni *Z*, ali to je još uvijek reprezentacija istog znaka. S druge strane, malo slovo z je definiran kao zaseban znak, koji pak opet može imati različite reprezentacije. Ovo je na kraju stvar definicije. U repertoaru znakova samo je jedno veliko slovo Z, ali moglo bi se definirati da su Z i z dva različita grafema istog elementa repertoara.


**FONTOVI**

Repertoar grafema sačinjava font. Odnosno, font je numerirani skup grafema. Brojevi odgovaraju kodnim pozicijama znakova (predstavljenih grafemima). Dakle, font je u tom smislu ovisan o kodu znakova. Izraz kao što je „Unicode font“ odnosi se na takva pitanja i ne implicira da font sadrži grafeme za sve znakove Unicodea. Može se dogoditi da font koji grafički predstavlja neki repertoar znakova nema različit grafem za svaki znak. Jedna važna napomena je da se ne smiju koristiti zamjenski znakovi koji samo izgledaju slično traženom znaku. Takve stvari zbunjuju alate za pretraživanje teksta, za provjeru pravopisa, sintetizatore govora, indeksere itd. Također ne možemo znati kako će znak izgledati pri promjeni fonta ili kodne stranice.



# 6. PRAKTIČNE NAPOMENE

***„There Ain't No Such Thing As Plain Text“***

Kad god se tekstualni podaci šalju preko mreže, pošiljatelj i primatelj moraju imati zajednički dogovor o korištenom kodnom sustavu za prikaz znakova. U optimalnom slučaju, softver se automatski brine o ovome, ali u praksi korisnici trebaju poduzeti neke mjere opreza.

Najvažnije, treba se uvjeriti da softver povezan s Internetom koji se koristi za prijenos podataka specificira kodiranje točno u odgovarajućim zaglavljima. Dvije su stvari u pitanju: zaglavlje mora postojati te mora odražavati stvarno kodiranje koje se koristi; i kodiranje mora biti takvo da je potpuno shvaćeno od strane softvera (potencijalnog) primatelja.

Korisno je saznati kako treba prilagoditi web pretraživač, poslužitelj vijesti i e-mail program tako da može prikazati podatke o kodiranju za stranicu, članak ili poruku koju čitamo. (Na primjer u Netscapeu se koristi „View Page Info“, u News Xpressu „View Raw Format“, u Pineu „h“.) Ako koristimo recimo Netscape za slanje e-maila ili objavljivanje poruke na Usenet vijestima, treba provjeriti da se poruka šalje u razumnom obliku. Konkretno, da ju ne šalje kao HTML ili ju duplicira slanjem kao obični tekst i kao HTML („plain text only“). U pogledu kodiranja znakova, trebalo bi se pobrinuti da je jedno od globalno prihvaćenih, kao što su to ASCII, neki od ISO 8859 ili UTF-8, ovisno o veličini repertoara znakova koja je potrebna. Tako u zaglavlju e-mail poruke možemo očekivati nešto kao

**Content-Type: text/plain; charset="UTF-8"**

a u zaglavlju web stranice

```
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8">
```

Posebno treba izbjegavati slanje podataka u kodiranju specifičnom za vlasnika, kao što su Macintosh i DOS kodovi, na javnu mrežu. U najgorem slučaju ako se to već radi, treba se uvjeriti da zaglavlje poruke specificira kodiranje. Nema problema kad se takvo kodiranje koristi unutar jednog računala ili u prijenosu podataka među sličnim računalima. Ali kad se šalje na Internet, program za slanje podatke treba konvertirati u šire prihvaćen oblik kodiranja. Ako ne možete podesiti program da to uradi, nabavite drugi program. Isto vrijedi kada se podaci prenose u digitalnom obliku na fizičkim medijima, npr. optičkim diskovima, disketama. Potrebno je poznavati program kojim su podaci stvoreni da bi se prepoznalo kodiranje. Ili se može primjenjivati metoda pokušaja i pogrešaka dok se za rezultat ne dobije nešto smisleno.

# 7. LITERATURA

http://spvp.zesoi.fer.hr/vjezbe/vj1_rs232/ASCII_tablica.htm
http://www.unicodecharacter.com/charsets/iso8859.html
http://www.unicode.org
http://scripts.sil.org/cms/scripts
http://web.archive.org/web/20030622083607/www.diffuse.org/chars.html
http://web.archive.org/web/20030618144134/www.diffuse.org/charguide.html
http://www.i18nguy.com/unicode/codepages.html