Računalniške komunikacije
in omrežja

Transportni sloj

Program INFORMATIKA
Višja strokovna šola Velenje
- dislocirana enota Murska Sobota

Prenosni sloj

- Prenosni (ali tudi transportni) sloj zaokroži protokole neposredno vezane na tehnologije v omrežju.
- Protokolna podatkovna enota (PDU) prenosnega sloja je **segment**.
- Prenosni sloj skrije podrobnosti delovanja omrežja in uporabnikom omogoči, da omrežje obravnavajo kot skupek med seboj povezanih računalnikov.

- Najpomembnejši funkciji sta:
  1. pošlji kos podatkov do oddaljenega vozlišča in
  2. vzpostavi zanesljivo povezavo do oddaljenega vozlišča.

Osnovni funkciji prenosnega sloja

- Če želimo samo **poslati kos podatkov do oddaljenega vozlišča** in nas zanesljivost prejema niti ne zanima, potem prenosni sloj nima veliko dela:
  - Na oddajni strani: podatke razdeli v primerno velike kose, jih odda omrežnemu sloju,
  - Na sprejemni strani: v primeru pravilnega prenosa kose spet sestavi v originalne podatke.
- Precej drugače pa je, če želimo **vzpostaviti zanesljivo povezavo** med dvema vozliščema.
  - Poseben primer je, če zanesljivo povezavo omogoča že omrežni sloj in prenosni sloj to lastnost izkoristi. V splošnem pa se prenosni sloj ne zanaša na zanesljivost omrežja, ampak sam poskrbi za ustrezne mehanizme.
  - Zagotavljanju zanesljivega prenosa otežkoča predvsem dejstvo, da se na poti skozi omrežje posamezni paketi lahko prehitevajo ali podvojijo oz. da se lahko določeni paketi nenormalno dolgo zakasnijo.

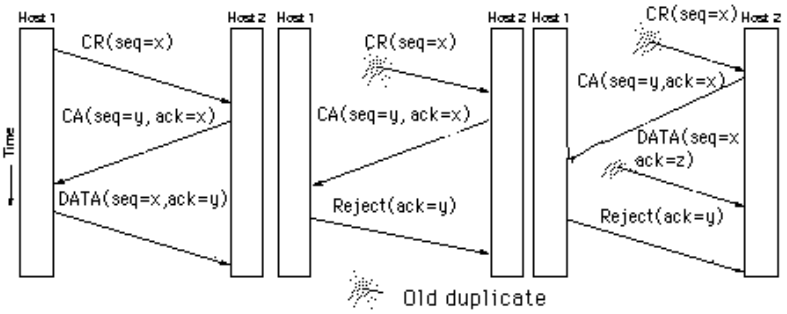
Številčenje

- Sistem zaporednih številk je nujno potreben za zanesljiv prenos.
- Številke se ob vsaki zahtevi ne smejo začeti od 0 naprej, ker bi se zaradi zakasnitev lahko promet iz dveh različnih zvez pomešal med seboj.
- Zaporedne številke se morajo torej vedno povečevati in se lahko vrnejo na 0 šele takrat, ko zagotovo ne bo v omrežju več nobenega starega paketa s to številko.
- Problem pa se pojavi, če eden od računalnikov izpade in se ponovno vrne v omrežje, pri tem pa pozabi, pri kateri številki je ostal.

Trikratno rokovanje

- Težava v zvezi s številkami je tudi ta, kako naj se vozlišči med seboj dogovorita, pri kateri številki bosta začela. Zanesljiva rešitev je uporaba TRIKRATNEGA ROKOVANJA (ang. three-way handshake), ki deluje na naslednji način:
  1. pošiljatelj pošlje svojo izbrano številko k naslovniku,
  2. naslovnik potrdi prejem in hkrati pošlje svojo izbrano številko,
  3. pošiljatelj potrdi prejem.

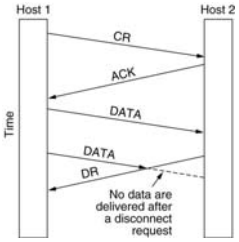
Trikratno rokovanje



CR (seq=x)– zahteva po povezavi (ang. connection request). Host 1 začne številčenje z x.
CA (seq=y, ack=x)– potrditev povezave (ang. connection accepted). Host 2 začne številčenje z y.
Reject – zavrnitev

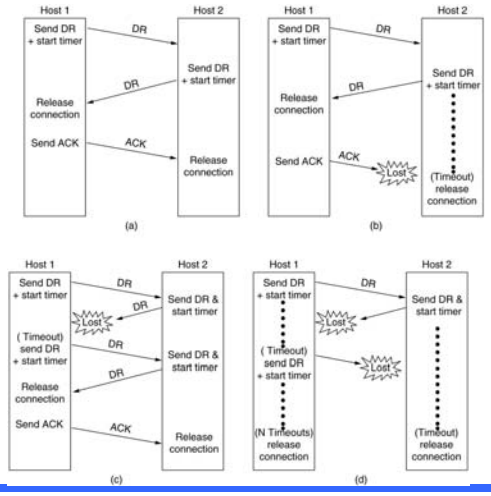
Končanje povezave

- Prenosni sloj ima težave tudi ob končanju povezave, saj se nadzor nad povezavo med vozliščema opravlja v obliki sporočil, ki pa se lahko tudi izgubijo.
- Preprečiti želimo, da bi eno vozlišče končalo povezavo, drugo pa o tem ne bi bilo obveščeno in bi zato v nedogled čakalo na podatke. Izkaže se, da je problem teoretično nerešljiv, v praksi pa rešitve temeljijo na uporabi **časovnikov**.



DR – zahteva po prekinitvi (ang. Disconection Request)

Končanje povezave



Naslavljanje v prenosnem sloju

- Isti transportni sloj služi večim aplikacijam naenkrat, zato je njegova naloga tudi to, da loči med seboj podatke namenjene različnim aplikacijam.
- Takšno ločevanje zahteva, da so podatki opremljeni z oznako aplikacije, ki so ji namenjeni. Poseben problem je, kako transportni sloj na osnovi te oznake, ki je ponavadi številka, prepozna aplikacijo.
  - Ena rešitev je, da ima pri sebi tabelo vseh pomembnejših aplikacij in njihovih oznak. Tak način si lahko privoščimo le za omejen nabor aplikacij.
  - Enako tabelo pa morajo uporabljati tudi vse druge naprave.

Prenosni sloj in aplikacijski sloj

- Ko prenosni sloj dobi podatke za določeno aplikacijo, ji dobljene podatke posreduje, kar pa lahko stori le, če je aplikacija aktivna. Če je število aplikacij v nekem vozlišču veliko, je potratno, da bi bile za vsak slučaj vseskozi vse aktivne.
- Učinkovitejša rešitev je, da imamo posebno aplikacijo, imenuje se procesni strežnik (ang. process server), ki sprejema podatke naslovljene na različne aplikacij in ob zahtevi za vzpostavitev povezave aktivira zahtevano aplikacijo.

Računalniške komunikacije
in omrežja

Transportni sloj v internetu

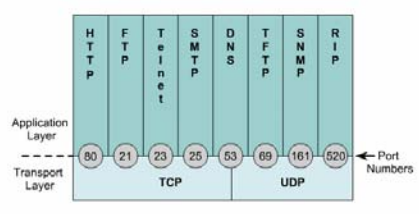
Program INFORMATIKA
Višja strokovna šola Velenje
- dislocirana enota Murska Sobota

Transportni sloj interneta

- Dva glavna protokola transportnega sloja interneta sta
  - povezavno usmerjen (ang. connection-oriented) **TCP**, če je zahtevan zanesljiv prenos podatkov
  - nepovezavni (ang. connectionless) **UDP**, če aplikacije le pošiljajo podatke brez vzpostavljene povezave.

Vrata

- Transportni sloj nudi storitve različnim aplikacijam, zato je v segmentu zapisana tudi **številka vrat** (ang. port).
- Vrata so 16-bitne številke.
- Številke manjše od 1024 so dobro znana vrata (ang. well-known ports)



UDP

- UDP (''User Datagram Protocol'') je preprost protokol transportnega sloja, ki od mehanizmov za zagotavljanje zanesljivosti vključuje le kontrolno vsoto.
- UDP protokol uporabljajo aplikacije, ki pošiljajo le eno zahtevo in en odgovor in zato ni potrebno vzpostavljati povezavo.
- Paket UDP vsebuje:
  - **Source port** (številki vrat na oddajni strani),
  - **Destination port** (številki vrat na sprejemni strani),
  - Length (dolžina paketa) in
  - **UDP checksum** (kontrolna vsota za glavo in podatke, ni obvezna) in
  - Data (podatki).

UDP header:

00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15	16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
Source Port	Destination Port
Length	Checksum
Data :::	

TCP

- TCP (''Transmission Control Protocol'') omogoča zanesljiv prenos podatkov med vozlišči.
- Pred prenosom podatkov se vzpostavi zanesljiva povezava.
- Povezava TCP je dvosmerna, podatki pa se izmenjujejo v obliki paketov, ki jih imenujemo SEGMENTI.
- Paket TCP je lahko tudi brez podatkov.
- Za kontrolo pretoka se uporablja protokol z drsečim oknom in natovarjanjem potrditev.

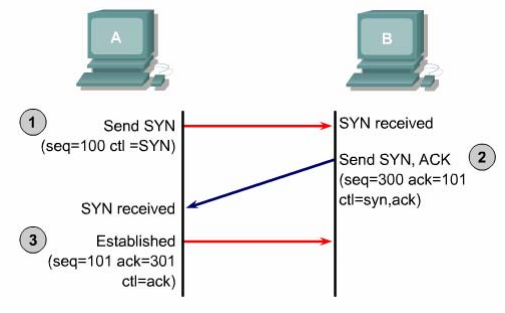
TCP

- Paket TCP je sestavljeno iz naslednjih polj:
  - **Source port** (številka vrat na oddajni strani),
  - **Destination port** (številka vrat na sprejemni strani),
  - **Sequence number** (zaporedna številka segmenta, za pravilno sestavljanje segmentov),
  - **Acknowledgement number** (številka potrditve, uporablja se natovarjanje potrditev),
  - Offset (odmik določa dolžino polj, ki sledijo),
  - Reserved (rezervirano),
  - **Code** (kode: syn, ack, fin itd),
  - **Window** (velikost paketa, ki ga je pošiljatelj pripravljen sprejeti),
  - **Checksum** (nadzorna vsota),
  - Urgent pointer (določa, kje se končajo nujni podatki),
  - Options (opcije) in
  - Data (podatki).

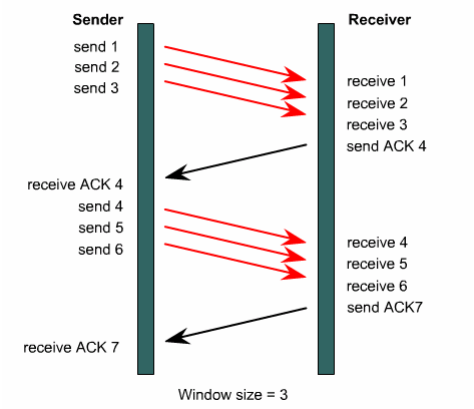
TCP header:

00 01 02 03 04 05 06 07 08 09 10 11 12 13 14 15	16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31
Source Port	Destination Port
Sequence Number	
Acknowledgment Number	
Data Offset / reserved / ECN / Control Bits	Window
Checksum	Urgent Pointer
Options and padding :::	
Data :::	

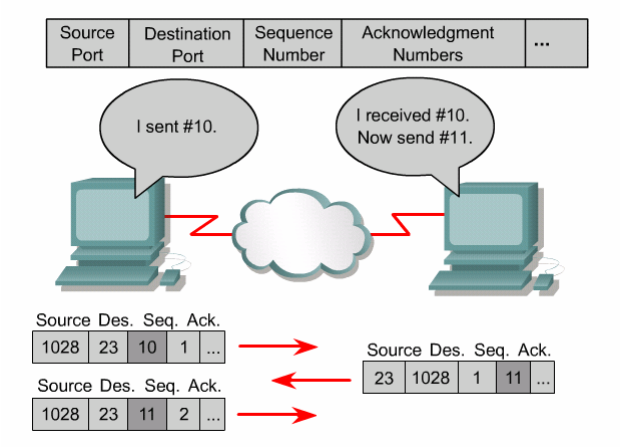
Trikratno rokovanje



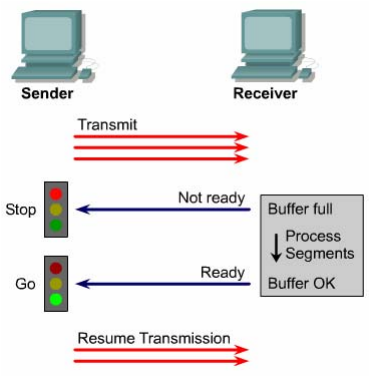
TCP drseče okno



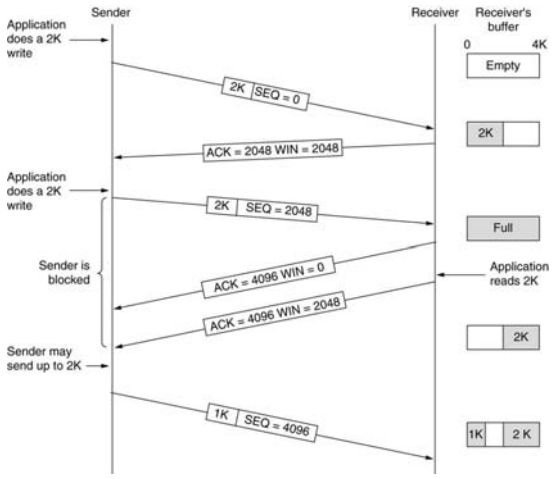
Natovarjanje potrditev



Upravljanje prenosa podatkov



Upravljanje prenosa podatkov



Računalniške komunikacije in omrežja

Podporne storitve

Program INFORMATIKA
Višja strokovna šola Velenje
- dislocirana enota Murska Sobota

Kaj so podporne storitve

- Podporne storitve vključujejo standardizirane mehanizme, ki jih sloji do vključno transportnega sloja ne zajemajo. Brez njih omrežja ne bi bila izvedljiva, oz. uporabna.
- V referenčnem modelu OSI sta podpornim storitvam namenjena sejni in predstavitveni sloj, a so številne med njimi kljub tipični "podporni" vlogi uvrščene tudi v aplikacijski sloj.
- Podporne storitve delimo na tiste, ki so namenjene:
  - logičnemu povezovanju med aplikacijskimi procesi (sejni sloj)
  - usklajevanju predstavitve podatkov (predstavitveni sloj).

Sejna povezava

- Sejni sloj omogoča LOGIČNO POVEZOVANJE aplikacijskih procesov.
- Tisti del komunikacije, ki ga transportni sloj obravnava kot eno celoto, imenujemo TRANSPORTNA POVEZAVA.
- Tisti del komunikacije, ki spada v isto sejo, pa imenujemo SEJNA POVEZAVA.
- Trajanje sejne povezavo je lahko bistveno drugačno od trajanja transportne povezave. Pri ISO-ju dopuščajo tri možnosti:
  - ena sejna povezava obsega eno transportno povezavo,
  - ena sejna povezava obsega več transportnih povezav,
  - ena transportna povezava obsega **več zaporednih** sejnih povezav.
- Izključena je možnost, da bi se več sej vzporedno prenašalo preko iste transportne povezave.

Nadzor poteka seje

- Eden od mehanizmov sejnega sloja so **PIŠKOTKI** (ang. cookies).
  - Ko se uporabnik z brskalnikom poveže na določeno stran in se tam prijavi, se na njegov računalnik namesti majhen paket podatkov imenovan piškotek.
  - Med nadaljno komunikacijo strežnik preverja, ali ima uporabnik nameščen ustrezen piškotek.
  - Piškotek lahko ostane na uporabnikovem računalniku dolgo časa in ves ta čas ga bo strežnik prepoznal (npr. brez ponovne prijave).

Predstavitveni sloj

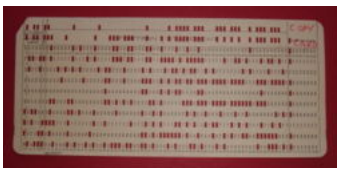
- Predstavitveni sloj skrbi za združljivo oz. enotno PREDSTAVITEV PODATKOV.
- Razlikujemo 5 večjih skupin:
  - zagotavljanje združljivosti predstavitve binarnih tipov podatkov,
  - zagotavljanje združljivosti predstavitve alfanumeričnih znakov,
  - zagotavljanje združljivosti predstavitve slik in multimedijskih vsebin,
  - stiskanje podatkov,
  - šifriranje podatkov.

Združljivost binarnih tipov podatkov

- Združlivost binarnih tipov podatkov je najtežji problem:
  - prevajalniki za različne programske jezike npr. na različen način predstavijo osnovne podatkovne strukture,
  - urejevalniki besedil npr. na različen način predstavijo oblikovanje besedila itd.

Združljivost alfanumeričnih tipov podatkov

- Predstavitev alfanumeričnih znakov temelji na uporabi KODNIH TABEL, v katerih za vsak znak piše njegova binarna koda.
- Ena prvih kodnih tabel za znake je bila **EBCDIC**, ki so jo vpeljali v podjetju IBM okoli leta 1964. Na luknjanih karticah so bili znaki kodirani z 8 biti.



- Naslednik EBCDIC je standard za kodiranje znakov **ASCII**, ki je nastal leta 1968, današnjo obliko pa je dobil leta 1986. Znaki so predstavljeni s 7 biti.

ASCII

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

Znaki so predstavljeni s 7 biti. Poleg črk so v tabeli še posebni znaki, ki so namenjeni različnim komunikacijskim protokolom (npr. ACK, NAK, STX in ETX) in podajanju strukture besedila (npr. nova vrstica).

ISO 8859

- Standard ISO 8859 je razširitev ASCII tabele na 256 znakov, torej je vsak znak predstavljen z 8 biti.
- Vsi ASCII znaki ostanejo na svojem mestu, na prosta mesta od 128 naprej pa se dodajo novi znaki. Glede na dodane znake je nastalo več različic kodiranja.

ISO 8859-1

- ISO 8859-1 je namenjena zahodnim evropskim državam in je znana tudi pod imenom **Latin-1**.
- V tabeli ISO 8859-1 so v primerjavi z ASCII dodatne črke, ki jih uporabljajo v Nemčiji, Franciji, Italiji, na Portugalskem, v skandinavskih državah in še v nekaterih drugih.
- Kodiranje npr. v celoti podpira albanščino in afriški jezik swahili, nima pa vseh črk za države srednje in vzhodne Evrope!

ISO/IEC 8859-1

	x0	x1	x2	x3	x4	x5	x6	x7	x8	x9	xA	xB	xC	xD	xE	xF
0x	*unused*															
1x																
2x	SP	!	"	#	$	%	&	'	(	)	*	+	,	-	.	/
3x	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4x	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5x	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6x	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7x	p	q	r	s	t	u	v	w	x	y	z	{	|	}	~	
8x	*unused*															
9x																
Ax	NBSP	¡	¢	£	¤	¥	¦	§	¨	©	ª	«	¬	SHY	®	¯
Bx	°	±	²	³	´	µ	¶	·	¸	¹	º	»	¼	½	¾	¿
Cx	À	Á	Â	Ã	Ä	Å	Æ	Ç	È	É	Ê	Ë	Ì	Í	Î	Ï
Dx	Ð	Ñ	Ò	Ó	Ô	Õ	Ö	×	Ø	Ù	Ú	Û	Ü	Ý	Þ	ß
Ex	à	á	â	ã	ä	å	æ	ç	è	é	ê	ë	ì	í	î	ï
Fx	ð	ñ	ò	ó	ô	õ	ö	÷	ø	ù	ú	û	ü	ý	þ	ÿ

ISO 8859-15

- Pozneje je bil ISO 8859-1 nadgrajen v ISO 8859-15, pri katerem so redko uporabljene simbole nadomestili z bolj potrebnimi, npr. simbolom za Euro. Dodali so tudi Š in Ž, na našo veliko žalost pa manjka črka Č.
- Standard ISO 8859-15 imenujemo tudi **Latin-9**.

ISO-8859-15

	x0	x1	x2	x3	x4	x5	x6	x7	x8	x9	xA	xB	xC	xD	xE	xF
0x	NUL	SOH	STX	ETX	EOT	ENQ	ACK	BEL	BS	HT	LF	VT	FF	CR	SO	SI
1x	DLE	DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	GS	RS	US
2x	SP	!	"	#	$	%	&	'	(	)	*	+	,	-	.	/
3x	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4x	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5x	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6x	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7x	p	q	r	s	t	u	v	w	x	y	z	{	|	}	~	DEL
8x	PAD	HOP	BPH	NBH	IND	NEL	SSA	ESA	HTS	HTJ	VTS	PLD	PLU	RI	SS2	SS3
9x	DCS	PU1	PU2	STS	CCH	MW	SPA	EPA	SOS	SGCI	SCI	CSI	ST	OSC	PM	APC
Ax	NBSP	¡	¢	£	€	¥	Š	§	š	©	ª	«	¬		®	¯
Bx	°	±	²	³	Ž	µ	¶	·	ž	¹	º	»	Œ	œ	Ÿ	¿
Cx	À	Á	Â	Ã	Ä	Å	Æ	Ç	È	É	Ê	Ë	Ì	Í	Î	Ï
Dx	Ð	Ñ	Ò	Ó	Ô	Õ	Ö	×	Ø	Ù	Ú	Û	Ü	Ý	Þ	ß
Ex	à	á	â	ã	ä	å	æ	ç	è	é	ê	ë	ì	í	î	ï
Fx	ð	ñ	ò	ó	ô	õ	ö	÷	ø	ù	ú	û	ü	ý	þ	ÿ

ISO-8859-1

- V standardu ISO 8859-1 je nekaj prostih mest, ki so jih pri ISO pozneje zapolnili s posebnimi znaki namenjenimi protokolom v internetu in tako je nastal standard ISO-8859-1 (dodatni pomišljaj).
- Pri Microsoftu so se odločili, da bodo v svojih operacijskih sistemih prav tako uporabili ISO 8859-1, a so prazna namesta napolnili z drugačnimi znaki. Njihovo kodiranje je znano pod oznako Windows-1252 oz. CP1252 in podobno kot Latin-1 **ne vsebuje slovenske črke Č**.

ISO-8859-1

	x0	x1	x2	x3	x4	x5	x6	x7	x8	x9	xA	xB	xC	xD	xE	xF
0x	NUL	SOH	STX	ETX	EOT	ENQ	ACK	BEL	BS	TAB	LF	VT	FF	CR	SO	SI
1x	DLE	DC1	DC2	DC3	DC4	NAK	SYN	ETB	CAN	EM	SUB	ESC	FS	GS	RS	US
2x	SP	!	"	#	$	%	&	'	(	)	*	+	,	-	.	/
3x	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4x	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5x	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6x	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7x	p	q	r	s	t	u	v	w	x	y	z	{	|	}	~	DEL
8x	PAD	HOP	BPH	NBH	IND	NEL	SSA	ESA	HTS	HTJ	VTS	PLD	PLU	RI	SS2	SS3
9x	DCS	PU1	PU2	STS	CCH	MW	SPA	EPA	SOS	SGCI	SCI	CSI	ST	OSC	PM	APC
Ax	NBSP	¡	¢	£	¤	¥	¦	§	¨	©	ª	«	¬	SHY	®	¯
Bx	°	±	²	³	´	µ	¶	·	¸	¹	º	»	¼	½	¾	¿
Cx	À	Á	Â	Ã	Ä	Å	Æ	Ç	È	É	Ê	Ë	Ì	Í	Î	Ï
Dx	Ð	Ñ	Ò	Ó	Ô	Õ	Ö	×	Ø	Ù	Ú	Û	Ü	Ý	Þ	ß
Ex	à	á	â	ã	ä	å	æ	ç	è	é	ê	ë	ì	í	î	ï
Fx	ð	ñ	ò	ó	ô	õ	ö	÷	ø	ù	ú	û	ü	ý	þ	ÿ

ISO 8859-2

- Srednji in vzhodni Evropi je namenjena kodna tabela ISO 8859-2, znana tudi pod imenom **Latin-2**.
- Pri Microsoftu so za jezike centralne Evrope definirali CP1250, ki vsebuje tudi vse slovenske šumnike. Zanj se je ustalilo ime MS-Latin2.

ISO/IEC 8859-2

	x0	x1	x2	x3	x4	x5	x6	x7	x8	x9	xA	xB	xC	xD	xE	xF
0x	*unused*															
1x																
2x	SP	!	"	#	$	%	&	'	(	)	*	+	,	-	.	/
3x	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4x	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5x	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6x	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7x	p	q	r	s	t	u	v	w	x	y	z	{	|	}	~	
8x	*unused*															
9x																
Ax	NBSP	Ą	˘	Ł	¤	Ľ	Ś	§	¨	Š	Ş	Ť	Ź		Ž	Ż
Bx	°	ą	˛	ł	´	ľ	ś	ˇ	¸	š	ş	ť	ź	˝	ž	ż
Cx	Ŕ	Á	Â	Ă	Ä	Ĺ	Ć	Ç	Č	É	Ę	Ë	Ě	Í	Î	Ď
Dx	Đ	Ń	Ň	Ó	Ô	Ő	Ö	×	Ř	Ů	Ú	Ű	Ü	Ý	Ţ	ß
Ex	ŕ	á	â	ă	ä	ĺ	ć	ç	č	é	ę	ë	ě	í	î	ď
Fx	đ	ń	ň	ó	ô	ő	ö	÷	ř	ů	ú	ű	ü	ý	ţ	˙

ISO 8859-16

- Pozneje so sprejeli še ISO 8859-16, ki je namenjena južni Evropi in prav tako vsebuje vse slovenske črke.
- Standard ISO 8859-16 imenujemo tudi **Latin-10**.
- Pri tej kodni tabeli so izpustili skorajda vse simbole in se trudili zajeti čimveč črk iz različnih jezikov.

ISO/IEC 8859-16															
	x0	x1	x2	x3	x4	x5	x6	x7	x8	x9	xA	xB	xC	xD	xE	xF
0x	unused															
1x																
2x	SP	!	"	#	$	%	&	'	(	)	*	+	,	-	.	/
3x	0	1	2	3	4	5	6	7	8	9	:	;	<	=	>	?
4x	@	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O
5x	P	Q	R	S	T	U	V	W	X	Y	Z	[	\	]	^	_
6x	`	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o
7x	p	q	r	s	t	u	v	w	x	y	z	{	|	}	~	
8x	unused															
9x																
Ax	NBSP	Ą	ą	Ł	€	„	Š	§	š	©	Ș	«	Ź		ź	Ż
Bx	°	±	Č	ł	Ž	"	¶	·	ž	č	ș	»	Œ	œ	Ÿ	ż
Cx	À	Á	Â	Ă	Ä	Ć	Æ	Ç	È	É	Ê	Ë	Ì	Í	Î	Ï
Dx	Đ	Ń	Ò	Ó	Ô	Ő	Ö	Ś	Ű	Ù	Ú	Û	Ü	Ę	Ț	ß
Ex	à	á	â	ă	ä	ć	æ	ç	è	é	ê	ë	ì	í	î	ï
Fx	đ	ń	ò	ó	ô	ő	ö	ś	ű	ù	ú	û	ü	ę	ț	ÿ

ISO 8859

- Še nekaj ostalih kodnih tabel iz družine ISO 8859:
  - ISO 8859-5 vsebuje znake iz cirilice,
  - ISO 8859-6 vsebuje arabsko pisavo,
  - ISO 8859-7 vsebuje grško pisavo,
  - ISO 8859-8 vsebuje hebrejsko pisavo,

Unicode

- Iz zmešnjave s kodnimi tabelami se je končno pojavila ideja o eni veliki kodni tabeli, ki bi zajela vse jezike na svetu. Ta projekt se imenuje Unicode.
- Unicode (vsaj na začetku) ni prinesel željene rešitve, pač pa stvari samo še bolj zapletel.
- Unicode namreč ni ena kodna tabela, ampak zbirka pravil za oblikovanje različno velikih tabel:
  - vsaka črka zaseda veliko pomnilnika (zakaj bi npr. v Evropi imeli v tabeli tudi vse kitajski pismenke).
- Med kodnimi tabelami v projektu Unicode sta trenutno uveljavljeni UTF-8 in UTF-16, ki pa se med seboj bistveno razlikujeta.

UTF-8 in UTF-16

- **V UTF-8** so v tabeli sicer črke vseh svetovnih jezikov, a imajo črke iz tabele ASCII kratko kodo dolžine 8 bitov, dodatne evropske črke imajo kodo dolžine 16 bitov, ostali za nas bolj eksotični simboli pa imajo kodo dolžine 24 ali 32 bitov.
  - Standard UTF-8 je npr. osnova operacijskega sistema Linux,
  - za UTF-8 je tudi zahtevano, da ga pravilno obravnavajo vsi protokoli v internetu.
- Tudi **UTF-16** omogoča zapis znakov vseh svetovnih jezikov, kode znakov pa so zaporedja 16 bitov dolgih števil. Vsi naši znaki so seveda predstavljeni le z enim številom dolžine 16 bitov.
  - UTF-16 so izbrali pri novejših operacijskih sistemih pri Microsoftu
  - UTF-16 uporablja za predstavitev besedila tudi programski jezik JAVA.

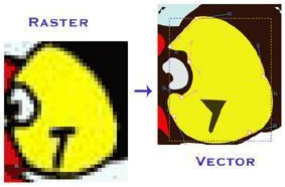
Združljivost predstavitve slik

- Združljivost predstavitve slik zagotovimo tako, da se dogovorimo za enoten format njihovega zapisa.
- Slike glede na predstavitev ločimo v dve skupini:
  - **VEKTORSKE** slike. Sestavljene iz osnovnih gradnikov kot pika, črta, krog, krivulja itd. Primerne so predvsem za skice in diagrame. Omogočajo poljubno pomanjševanje in povečevanje slike brez izgube kvalitete. Zelo znana formata za vektorsko predstavitev dokumentov sta POSTSCRIPT in PDF (danes podpira obe predstavitvi) .
  - **BITNE** (tudi rastrske) slike. Pri bitnih slikah je osnova za predstavitev matrika točk, vsaka točka pa ima določeno barvo. Matrika pik je zaradi varčevanja s prostorom na nek način stisnjena. Uporabljeno je lahko brezizgubno stiskanje (npr. BMP, PNG, GIF in osnovna različica TIFF) ali pa izgubno stiskanje (npr. JPEG).

Združljivost predstavitve slik

- Vektorska slika (GIF)

- Pretvorba bitne slike v vektorsko:



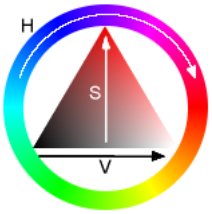
Barvni modeli

- Pri predstavitvi slik se uporabljajo različni BARVNI MODELI, ki določajo, iz katerih komponent so sestavljene posamezne barve. Znani barvni modeli so:
  - RGB (rdeča, zelena in modra).Uporabljajo ga monitorji.
  - CMYK (odtenek modre, odtenek vijoličaste, rumena in črna).Uporabljajo ga tiskalniki.
  - HSV (odtenek, nasičenost in svetlost).

RGB:

CMYK:

HSV:



Združljivost predstavitve multimedijskih vsebin

- Pri multimedijskih vsebinah moramo med drugim predstaviti zvok in video.
- Za predstavitev poskrbi cela množica algoritmov, vsakega od njih pa po analogiji s kodno tabelo imenujemo **KODEK**.
- Najpomembnejši standard za predstavitev multimedijskih vsebin je MPEG.
- Osnovna različica je **MPEG-1**, pri katerem se multimedijska vsebina razdeli na sloje:
  - sloj 1 je za sinhronizacijo,
  - sloj 2 je za predstavitev video signala,
  - sloj 3 je za predstavitev audio signala.
- Drugi in tretji sloj se lahko uporabita tudi vsak posebej, sloj 2 je znan kot format **Video CD**, sloje 3 pa kot format **MP3**.

Multimedijske vsebine

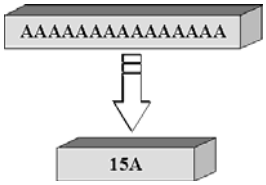
- **MPEG-2** je naslednik MPEG-1. Uporablja se za zapis na DVD in pri digitalni televiziji. Pri predstavitvi audio signala npr. dodaja možnost večkanalnega zvoka.
- Naslednik obeh standardov pa je **MPEG-4**, ki dodatno zagotavlja učinkovito stiskanje podatkov in zaščito avtorskih pravic. Na osnovi mehanizmov v MPEG-4 je bilo razvitih tudi več neuradnih različic, med katerimi je najbolj znana **DivX**.
- Obstaja tudi predstavitev zvoka, pri katerem ne uporabimo vzorčenja signala, ampak zvok predstavimo z notnim zapisom, tako kot glasbeniki. Primer take predstavitve je format **MIDI**, ki se uporablja npr. pri melodijah za zvonenje telefonov.

Stiskanje podatkov

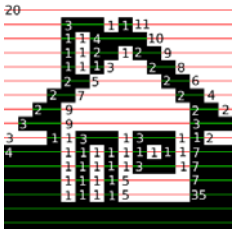
- Stiskanje podatkov je pomemben del vsakega omrežja, saj stisnjeni podatki manj obremenjujejo omrežje in s tem omogočajo večjo prepustnost. Ločimo:
  - **BREZIZGUBNO STISKANJE**. Pri brezizgubnem stiskanju lahko z ustreznim raztezanjem vedno v celoti povrnemo originalne podatke.
  - **IZGUBNO STISKANJE**. Po raztezanju dobimo le bolj ali manj dober približek originalnih podatkov. Uporabno je pri bitnih slikah in multimedijskih vsebinah, saj lahko z njim podatke bistveno bolj stisnemo, seveda pa pri tem žrtvujemo nekaj kvalitete.

RLE - brezizgubno stiskanje

- Zelo preprosto brezizgubno stiskanje je metoda **RLE**, ki temelji na učinkoviti predstavitvi enakih zaporednih simbolov.
- Metodo RLE lahko uporabimo tudi pri binarnih podatkih in je pogosto prisotna kot del bolj učinkovitih metod stiskanja.

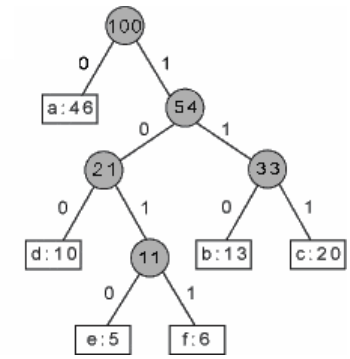


Dolžino smo zmanjšali na petino.



Huffmanov kod - brezizgubno stiskanje

- Učinkovito metodo stiskanja dobimo, če ne kodiramo vseh znakov s kodo enake dolžine, ampak za tiste bolj pogoste uporabimo krajšo kodo, za tiste bolj redke pa daljšo. Ta ideja je uporabljena že pri Morsejevi abecedi, v povezavi s stiskanjem podatkov pa je najbolj znana izvedba HUFFMANOV KOD.



LZ77 - brezizgubno stiskanje

- Podatke razdelimo na osnovne simbole, ki jih pri stiskanju obravnavamo v čim daljših zaporedjih. Za vsako zaporedje pogledamo, če se je v preteklosti že pojavilo. V tem primeru namesto zaporedja navedemo le dolžino zaporedja in mesto, kje v preteklosti se je to zaporedje že pojavilo.

The **receiver** requires a **receipt** for it. This **is** automatically sent when it **is** received.

The receiver#9#3quires a#20#6pt for it. This is automatically sent when it #30#3#47#6ved.

#m#n (m pove koliko znakov nazaj, n pa pove dolžino zaporedja)

Stiskanje podatkov

- Kombinacija algoritma LZ77 in Huffmanovega koda, ki jo imenujemo metoda **DEFLATE**, je uporabljena pri formatih ZIP in GZIP, ki sta trenutno najbolj razširjena načina stiskanja podatkov.
- Format za predstavitev slik GIF npr. uporablja različico algoritma LZ77, ki jo imenujemo **LZW**.
- Format PNG uporablja metodo DEFLATE.
- Veliko formatov za predstavitev slik in multimedije, med njimi npr. JPEG in MP3, uporablja Huffmanov kod.

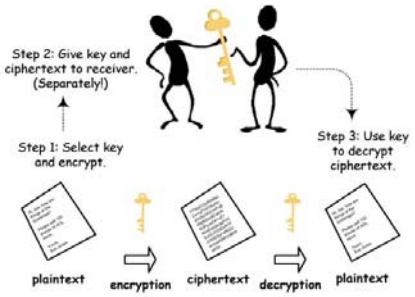
Šifriranje podatkov

- Osnovni namen šifriranja je skrivanje podatkov. Šifriranje se ne ukvarja s preprečevajem prisluškovanja in prestrezanja podatkov, ampak z zaščito informacij, ki jih nosijo podatki.
- Šifriranje imenujemo tudi KRIPTIRANJE, šifrirano sporočilo pa KRIPTOGRAM.
- Postopek šifriranja s tujko označimo kot ENKRIPCIJA.
- Obratni postopek (dešifriranje), ko iz kriptograma razberemo originalno sporočilo, pa je DEKRIPCIJA.

Simetrično šifriranje

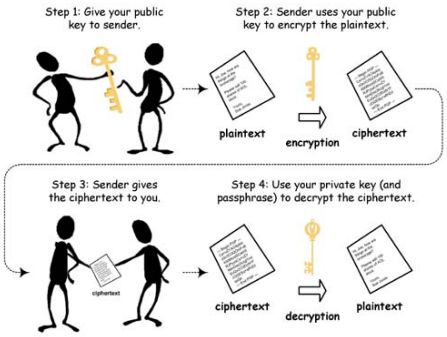
- Pri simetričem šifriranju sta enkripcijski in dekripcijski ključ med seboj povezana tako, da kdor pozna enega, pozna tudi drugega, zato morata biti oba skrivna.

  Simetrično šifriranje je hitro.
  - Primer: DES algoritem



Asimetrično šifriranje

- Pri asimetričnem šifriranju dekripcijskega ključa (imenujemo ga OSEBNI KLJUČ (ang. private key)) ne moremo enostavno pridobiti iz enkripcijskega (imenujemo ga JAVNI KLJUČ (ang. public key)), zato enkripcijskega ključa ni potrebno skrivati.
  Asimetrično šifriranje je počasno.
  - Primer: RSA algoritem



Varnost v omrežjih

- Ko govorimo o varnosti omrežja, mislimo na naslednje zahteve:
  - **ZAUPNOST** podatkov: podatke lahko vidijo le pooblaščene osebe,
  - **CELOVITOST** podatkov (integriteta, neokrnjenost): sprejeti podatki so enaki oddanim,
  - **VERODOSTOJNOST** podatkov (pristnost, avtentičnost): podatke je v resnici oddal naveden oddajnik,
  - **RAZPOLOŽLJIVOST** omrežja: omrežje je vedno na voljo za komunikacijo,
  - **OMEJITEV DOSTOPA** do omrežja: omrežje ni na razpolago nepooblaščenim osebam.
- Glede na podane zahteve lahko napade na omrežje razvrstimo po učinku:
  - prestrezanje podatkov je napad na zaupnost podatkov,
  - spreminjanje podatkov je napad na celovitost podatkov,
  - ponarejanje podatkov o pošiljatelju je napad na verodostojnost podatkov,
  - onemogočanje komunikacije je napad na razpoložljivost omrežja,