Quaid-e-Awam University of Engineering, Science and Technology

بسم الله الرحمن الرحيم

In the name of Allah, the Most Merciful, the Most Kind

Date: 22-09-2021

BCS 103
Digital Logic & Computer Architecture

Lecture 9 and 10

Department of Computer Science

Classification of binary codes

The codes are broadly categorized into following four categories.

- **Weighted Codes**
  - BCD (8421)
  - 6311
  - 2421
  - 642-3
  - 84-2-1
- **Non-Weighted Codes**
  - Excess-3 Codes
  - Gray Codes
- **Alphanumeric Codes**
  - ASCII
  - EBCDIC
- **Error Detecting Codes (Parity)**

Done

Today

Alphanumeric Codes / Characters

Alphanumeric characters are used to make words and strings. They include uppercase and lowercase letters, the digits 0-9, and symbols such as ? + and £.

Computers are unable to process these characters directly as they only process binary code. So they need a way of converting these characters to binary code and vice versa. They can do this using character sets.

Character sets are collections of characters that a computer recognises from their binary representation.

Alphanumeric Codes / Characters

As well as the alphanumeric characters mentioned above, character sets also contain special characters which do certain commands (e.g. space, delete, enter).

So when you press a button on your keyboard it sends a binary signal to the computer telling it which key you pressed. The computer then uses the character set to translate the binary code into a particular character.

What is a character set?

A character set is a defined list of characters recognised by the computer hardware and software.

Alphanumeric Codes / Characters

ASCII is the most commonly-used character set in the English-speaking world. Each ASCII character is given a 7-bit binary code – this means that it can represent a total of 128 different characters, including all the letters in the English alphabet, numbers, symbols and commands.

An extra bit (0) is added to the start of the binary code for each ASCII character. This means that each ASCII character fits nicely into 1 byte (8 bits).

Alphanumeric Codes / Characters

Extended ASCII is a character set which gives each character an 8-bit binary code, allowing for 256 characters to be represented. The first 128 characters are in exactly the same order as the ASCII characters.

Extended ASCII is particularly useful for many European languages like French and German which include accents on some of the vowels, like á, Ô, and Ü.

ASCII

- ASCII code represents alphanumeric data in most computers.
- Stands for: ***"American Standard Code for Information Interchange"***
- It works like any other code. One thing represents another.
- In ASCII, binary is used to represent our numbers, letters and symbols.
- ASCII includes definitions for 128 characters:
- 33 are non-printing control characters (many now obsolete) that affect how text and space is processed and 95 printable characters, including space.

ASCII: Groupings

The ASCII codes are grouped as follows:

0 - 32	Control codes (non-printing)
33 - 47	Printable symbols such as ! / \ &
48 - 57	The digits 0-9
58 - 64	Printable symbols such as < > =
65 - 90	Upper case characters A to Z
91 - 96	Printable characters including [ ]
97 - 122	Lower case characters a to z
123 - 127	Printable characters including { }

ASCII

Non-printable control codes

Decimal	Hex	Char	Decimal	Hex	Char	Decimal	Hex	Char	Decimal	Hex	Char
0	0	[NULL]	32	20	[SPACE]	64	40	@	96	60	`
1	1	[START OF HEADING]	33	21	!	65	41	A	97	61	a
2	2	[START OF TEXT]	34	22	"	66	42	B	98	62	b
3	3	[END OF TEXT]	35	23	#	67	43	C	99	63	c
4	4	[END OF TRANSMISSION]	36	24	$	68	44	D	100	64	d
5	5	[ENQUIRY]	37	25	%	69	45	E	101	65	e
6	6	[ACKNOWLEDGE]	38	26	&	70	46	F	102	66	f
7	7	[BELL]	39	27	'	71	47	G	103	67	g
8	8	[BACKSPACE]	40	28	(	72	48	H	104	68	h
9	9	[HORIZONTAL TAB]	41	29	)	73	49	I	105	69	i
10	A	[LINE FEED]	42	2A	*	74	4A	J	106	6A	j
11	B	[VERTICAL TAB]	43	2B	+	75	4B	K	107	6B	k
12	C	[FORM FEED]	44	2C	,	76	4C	L	108	6C	l
13	D	[CARRIAGE RETURN]	45	2D	-	77	4D	M	109	6D	m
14	E	[SHIFT OUT]	46	2E	.	78	4E	N	110	6E	n
15	F	[SHIFT IN]	47	2F	/	79	4F	O	111	6F	o
16	10	[DATA LINK ESCAPE]	48	30	0	80	50	P	112	70	p
17	11	[DEVICE CONTROL 1]	49	31	1	81	51	Q	113	71	q
18	12	[DEVICE CONTROL 2]	50	32	2	82	52	R	114	72	r
19	13	[DEVICE CONTROL 3]	51	33	3	83	53	S	115	73	s
20	14	[DEVICE CONTROL 4]	52	34	4	84	54	T	116	74	t
21	15	[NEGATIVE ACKNOWLEDGE]	53	35	5	85	55	U	117	75	u
22	16	[SYNCHRONOUS IDLE]	54	36	6	86	56	V	118	76	v
23	17	[ENG OF TRANS. BLOCK]	55	37	7	87	57	W	119	77	w
24	18	[CANCEL]	56	38	8	88	58	X	120	78	x
25	19	[END OF MEDIUM]	57	39	9	89	59	Y	121	79	y
26	1A	[SUBSTITUTE]	58	3A	:	90	5A	Z	122	7A	z
27	1B	[ESCAPE]	59	3B	;	91	5B	[	123	7B	{
28	1C	[FILE SEPARATOR]	60	3C	<	92	5C	\	124	7C	|
29	1D	[GROUP SEPARATOR]	61	3D	=	93	5D	]	125	7D	}
30	1E	[RECORD SEPARATOR]	62	3E	>	94	5E	^	126	7E	~
31	1F	[UNIT SEPARATOR]	63	3F	?	95	5F	_	127	7F	[DEL]

Converting string characters to ASCII code

To a computer, a character in a string is just a number; a number representing one of the characters in the ASCII code. An algorithm might need to find the ASCII code for a character. All programming languages have commands for this.

Converting string characters to ASCII code

CHR() to return a character from a number.

ASC() to return a number from a character.

Therefore:

myString = CHR (67)

Print (myString)

Would return the letter upper case C as 67 is the ASCII code that letter.

Entering:

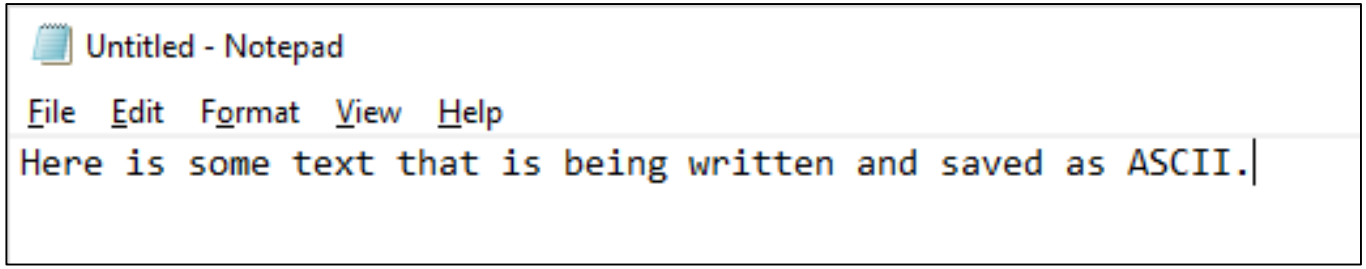
ASC (‘D’)

Would return the number 68 as that is its number in the ASCII code.

Size of ASCII files

Because on byte is used for each character, the size of a plain text file in bytes should be equal to the number of characters.

Example:



The file consist of 59 characters, including spaces, and so the size of the file should be 59 bytes.

E B C D I C

EBCDIC

Extended binary coded decimal interchange code (EBCDIC) is an 8-bit binary code for numeric and alphanumeric characters. It was developed and used by IBM. It is a coding representation in which symbols, letters and numbers are presented in binary language.

EBCDIC

EBCDIC is an 8-bit character encoding widely used in IBM midrange and mainframe computers. This encoding was developed in 1963 and 1964.

EBCDIC was developed to enhance the existing capabilities of binary-coded decimal code. This code is used in text files of S/390 servers and OS/390 operating systems of IBM.

Thanks