

TUGAS MODUL 1

PRAKTIKUM SISTEM OPERASI

Nama : Annisa Nugraheni

NIM : L200180066

Kelas : B

ASCII singkatan dari American Standard Code for Information Interchange atau Kode Standar Amerika untuk Pertukaran Informasi merupakan suatu standar internasional dalam kode huruf dan symbol seperti Hex dan Unicode tetapi ASCII lebih bersifat universal, contohnya 124 adalah untuk karakter “|”. Ia selalu digunakan oleh komputer dan alat komunikasi lain untuk menunjukkan teks. Kode ASCII sebenarnya memiliki komposisi bilangan biner sebanyak 7 bit. Namun, ASCII disimpan sebagai sandi 8 bit dengan menambahkan satu angka 0 sebagai bit signifikan paling tinggi. Bit tambahan ini sering digunakan untuk uji paritas. Karakter control pada ASCII dibedakan menjadi 5 kelompok sesuai dengan penggunaan, yaitu berturut-turut meliputi logical communication, device control, information separator, code extension, dan physical communication. Code ASCII ini banyak dijumpai pada papan ketik (keyboard) komputer atau instrumen-instrumen digital.

TABEL ASCII:

Decimal	Octal	Hex	Binary	Value	Description
000	000	00	0000 0000	NUL	"null" character
001	001	01	0000 0001	SOH	start of header
002	002	02	0000 0010	STX	start of text
003	003	03	0000 0011	ETX	end of text
004	004	04	0000 0100	EOT	end of transmission
005	005	05	0000 0101	ENQ	enquiry

Decimal	Octal	Hex	Binary	Value	Description
006	006	06	0000 0110	ACK	acknowledgment
007	007	07	0000 0111	BEL	bell
008	010	08	0000 1000	BS	backspace
009	011	09	0000 1001	HT	horizontal tab
010	012	0A	0000 1010	LF	line feed
011	013	0B	0000 1011	VT	vertical tab
012	014	0C	0000 1100	FF	form feed
013	015	0D	0000 1101	CR	carriage return
014	016	0E	0000 1110	SO	shift out
015	017	0F	0000 1111	SI	shift in
016	020	10	0001 0000	DLE	data link escape
017	021	11	0001 0001	DC1	device control 1 (XON)
018	022	12	0001 0010	DC2	device control 2
019	023	13	0001 0011	DC3	device control 3 (XOFF)
020	024	14	0001 0100	DC4	device control 4
021	025	15	0001 0101	NAK	negative acknowledgement
022	026	16	0001 0110	SYN	synchronous idle

Decimal	Octal	Hex	Binary	Value	Description
023	027	17	0001 0111	ETB	end of transmission block
024	030	18	0001 1000	CAN	cancel
025	031	19	0001 1001	EM	end of medium
026	032	1A	0001 1010	SUB	substitute
027	033	1B	0001 1011	ESC	escape
028	034	1C	0001 1100	FS	file separator
029	035	1D	0001 1101	GS	group separator
030	036	1E	0001 1110	RS	request to send/record separator
031	037	1F	0001 1111	US	unit separator
032	040	20	0010 0000	SP	space
033	041	21	0010 0001	!	exclamation mark
034	042	22	0010 0010	"	double quote
035	043	23	0010 0011	#	number sign
036	044	24	0010 0100	\$	dollar sign
037	045	25	0010 0101	%	percent
038	046	26	0010 0110	&	ampersand
039	047	27	0010 0111	'	single quote

Decimal	Octal	Hex	Binary	Value	Description
040	050	28	0010 1000	(left/opening parenthesis
041	051	29	0010 1001)	right/closing parenthesis
042	052	2A	0010 1010	*	asterisk
043	053	2B	0010 1011	+	plus
044	054	2C	0010 1100	,	comma
045	055	2D	0010 1101	-	minus or dash
046	056	2E	0010 1110	.	dot
047	057	2F	0010 1111	/	forward slash
048	060	30	0011 0000	0	
049	061	31	0011 0001	1	
050	062	32	0011 0010	2	
051	063	33	0011 0011	3	
052	064	34	0011 0100	4	
053	065	35	0011 0101	5	
054	066	36	0011 0110	6	
055	067	37	0011 0111	7	
056	070	38	0011 1000	8	

Decimal	Octal	Hex	Binary	Value	Description
057	071	39	0011 1001	9	
058	072	3A	0011 1010	:	colon
059	073	3B	0011 1011	;	semi-colon
060	074	3C	0011 1100	<	less than
061	075	3D	0011 1101	=	equal sign
062	076	3E	0011 1110	>	greater than
063	077	3F	0011 1111	?	question mark
064	100	40	0100 0000	@	"at" symbol
065	101	41	0100 0001	A	
066	102	42	0100 0010	B	
067	103	43	0100 0011	C	
068	104	44	0100 0100	D	
069	105	45	0100 0101	E	
070	106	46	0100 0110	F	
071	107	47	0100 0111	G	
072	110	48	0100 1000	H	
073	111	49	0100 1001	I	

Decimal	Octal	Hex	Binary	Value	Description
074	112	4A	0100 1010	J	
075	113	4B	0100 1011	K	
076	114	4C	0100 1100	L	
077	115	4D	0100 1101	M	
078	116	4E	0100 1110	N	
079	117	4F	0100 1111	O	
080	120	50	0101 0000	P	
081	121	51	0101 0001	Q	
082	122	52	0101 0010	R	
083	123	53	0101 0011	S	
084	124	54	0101 0100	T	
085	125	55	0101 0101	U	
086	126	56	0101 0110	V	
087	127	57	0101 0111	W	
088	130	58	0101 1000	X	
089	131	59	0101 1001	Y	
090	132	5A	0101 1010	Z	

Decimal	Octal	Hex	Binary	Value	Description
091	133	5B	0101 1011	[left/opening bracket
092	134	5C	0101 1100	\	back slash
093	135	5D	0101 1101]	right/closing bracket
094	136	5E	0101 1110	^	caret/circumflex
095	137	5F	0101 1111	_	underscore
096	140	60	0110 0000	`	
097	141	61	0110 0001	a	
098	142	62	0110 0010	b	
099	143	63	0110 0011	c	
100	144	64	0110 0100	d	
101	145	65	0110 0101	e	
102	146	66	0110 0110	f	
103	147	67	0110 0111	g	
104	150	68	0110 1000	h	
105	151	69	0110 1001	i	
106	152	6A	0110 1010	j	
107	153	6B	0110 1011	k	

Decimal	Octal	Hex	Binary	Value	Description
108	154	6C	0110 1100	l	
109	155	6D	0110 1101	m	
110	156	6E	0110 1110	n	
111	157	6F	0110 1111	o	
112	160	70	0111 0000	p	
113	161	71	0111 0001	q	
114	162	72	0111 0010	r	
115	163	73	0111 0011	s	
116	164	74	0111 0100	t	
117	165	75	0111 0101	u	
118	166	76	0111 0110	v	
119	167	77	0111 0111	w	
120	170	78	0111 1000	x	
121	171	79	0111 1001	y	
122	172	7A	0111 1010	z	
123	173	7B	0111 1011	{	left/opening brace
124	174	7C	0111 1100		vertical bar

Decimal	Octal	Hex	Binary	Value	Description
125	175	7D	0111 1101	}	right/closing brace
126	176	7E	0111 1110	~	tilde
127	177	7F	0111 1111	DEL	delete

Daftar perintah Bahasa assembly:

Instruksi	Keterangan Singkatan
ACALL	Absolute Call
ADD	Add
ADDC	Add with Carry
AJMP	Absolute Jump
ANL	AND Logic
CJNE	Compare and Jump if Not Equal
CLR	Clear
CPL	Complement
DA	Decimal Adjust
DEC	Decrement
DIV	Divide
DJNZ	Decrement and Jump if Not Zero
INC	Increment
JB	Jump if Bit Set
JBC	Jump if Bit Set and Clear Bit
JC	Jump if Carry Set
JMP	Jump to Address
JNB	Jump if Not Bit Set
JNC	Jump if Carry Not Set
JNZ	Jump if Accumulator Not Zero
JZ	Jump if Accumulator Zero
LCALL	Long Call
LJMP	Long Jump
MOV	Move from Memory
MOVC	Move from Code Memory
MOVB	Move from Extended Memory
MUL	Multiply
NOP	No Operation
ORL	OR Logic
POP	Pop Value From Stack
PUSH	Push Value Onto Stack
RET	Return From Subroutine

RETI	Return From Interrupt
RL	Rotate Left
RLC	Rotate Left through Carry
RR	Rotate Right
RRC	Rotate Right through Carry
SETB	Set Bit
SJMP	Short Jump
SUBB	Subtract With Borrow
SWAP	Swap Nibbles
XCH	Exchange Bytes
XCHD	Exchange Digits
XRL	Exclusive OR Logic