



Corso di Informatica B

prof. Renato Martucci

Meccanici ed Energetici

A.A. 2001-02

Libri del Corso

- **Ceri, Mandrioli, Sbattella - Informatica arte e mestiere - Mc Graw-Hill**
- **Bellini, Guidi - Linguaggio C, guida alla programmazione - Mc Graw-Hill**
- **Lucidi delle lezioni --> sul sito**
 - **<http://www.elet.polimi.it/Users/DEI/Sections/CompEng/Renato.Martucci/index.html>**

Approfondimenti

- **H.M. Deitel, P. J. Deitel, C corso completo di programmazione, Apogeo, 2000**
- **Tricky C - sito contenente un libro on-line su C**
 - <http://host.aspide.it/trickyc/trkcfm.htm>
- **A.Morzenti, S.Gaburri "Dispensa su ANSI C", file in formato pdf**
 - ftp://ftp.elet.polimi.it/outgoing/William.Fornaciari/www/docs/didattica/InfoB2001/dispense/dispensa_ANSI_C.pdf
- **Bellettini, Filippi, Sbattella, "Linguaggio C: Esercizi e temi d'esame risolti", Il ed, Prog. Leonardo, Esculapio, Bologna, 2000**

L'esame assegna 34 punti che corrispondono al 30 e lode, suddividendoli nel seguente modo:

- il laboratorio è parte integrante del corso e come tale concorre alla formazione del voto dell'esame. non ne è previsto il recupero. se lo studente non frequenta (almeno 4 sessioni di laboratorio), oppure ottiene una valutazione negativa (da 0 a 2 punti) da parte del responsabile del laboratorio, dovrà ripetere il corso nell'anno accademico successivo. Inoltre le prove in itinere sostenute saranno annullate.
- L'attività di laboratorio permette quindi di ottenere un punteggio "utile" (laboratorio superato) da 3 a 10 punti che sarà sommato al voto delle prove in itinere. La valutazione terrà conto della capacità del singolo studente di impostare la soluzione del problema in termini teorici e di realizzare la soluzione funzionante al calcolatore; secondariamente si considera l'impegno e la serietà dimostrata dallo studente durante le attività di laboratorio.
- Sono previste due prove in itinere nelle quali si assegna un massimo di 12 punti. Una prova in itinere è ritenuta superata se ottiene 7 punti, altrimenti deve essere recuperata.
- Se lo studente è insufficiente in una o tutte 2 le prove in itinere deve recuperarle nella stessa sessione o a settembre.
- Se uno studente supera tutte le prove (laboratorio compreso), totalizzando tuttavia un voto inferiore a 18 deve provvedere a recuperare una o entrambe le prove in itinere sostenute.
 - Esempi: $7 + 7 + 3\text{lab} \rightarrow$ deve recuperare una delle 2 prove in itinere a scelta
 - $6 + 7 + 3\text{lab} \rightarrow$ deve recuperare la prova insufficiente

Il Calcolatore

I processori

CPU (Central Process Unit) è il cervello del calcolatore ed ha il compito di eseguire i programmi immagazzinati nella memoria principale, prendendone le istruzioni, esaminandole ed eseguendole una dopo l'altra.

É costituita da:

- unità di controllo;
- unità aritmetico - logica;
- registri.

Il Calcolatore

Esecuzione delle istruzioni

- La CPU esegue le istruzioni attraverso una serie di piccoli passi:
 - legge la nuova istruzione dalla memoria ponendola nel registro delle istruzioni (IR);
 - cambia il contatore di programma affinché punti alla istruzione seguente;
 - determina il tipo di istruzione appena prelevata;
 - se l'istruzione usa dei dati in memoria, determina dove sono situati;
 - va a prendere i dati nei registri interni della CPU;
 - esegue l'istruzione;
 - memorizza il dato al posto giusto;
 - ricomincia il ciclo.
- **É una sequenza del tipo preleva-decodifica-esegui.**

Cosa e' un programma

Una sequenza di operazioni eseguite dalla macchina secondo un flusso predefinito.

Una “scatola” che dato un input di qualche tipo fornisce un output.

Un ***“automa a stati finiti”***.

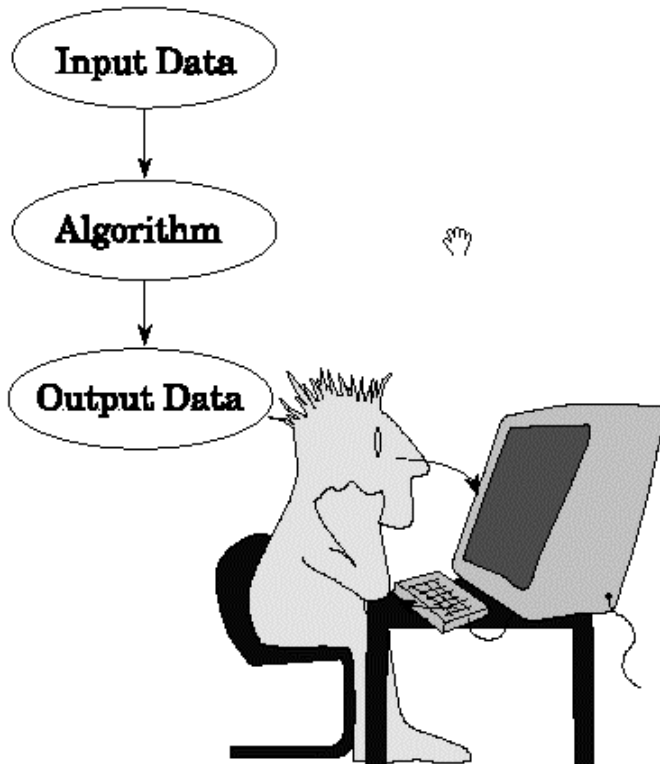
Cosa esegue un programma

Quello che il programmatore ha deciso di fargli fare.

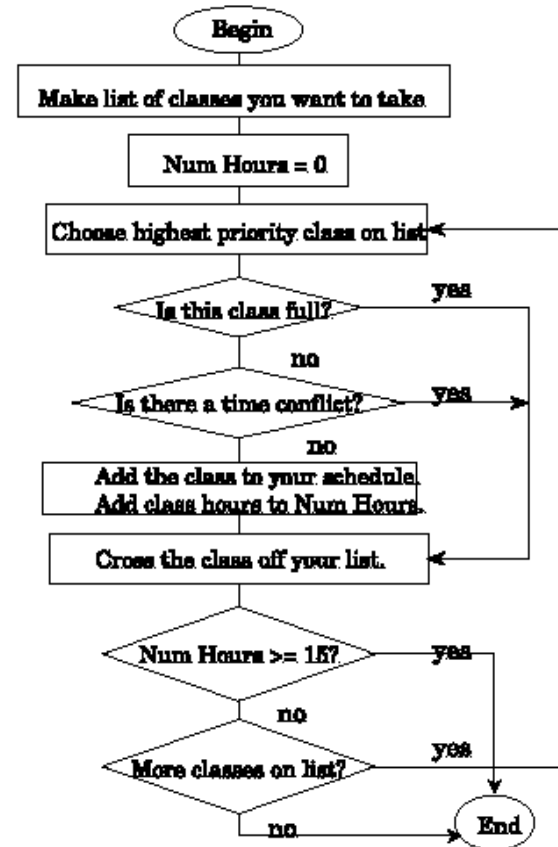
Un Algoritmo.

L'algoritmo

Algorithms in Computing



Flowcharts



Di cosa è fatto un programma

Un insieme “classificato” di operazioni

Istruzioni del linguaggio scritte
sequenzialmente

Livelli di Rappresentazione

High Level Language

Compilatore

Assembly Language

Assemblatore

Machine Language

*Interpretazione della
Macchina*

Segnali di Controllo

temp = v[k];

v[k] = v[k+1];

v[k+1] = temp;

lw \$15, 0(\$2)

lw \$16, 4(\$2)

sw \$16, 0(\$2)

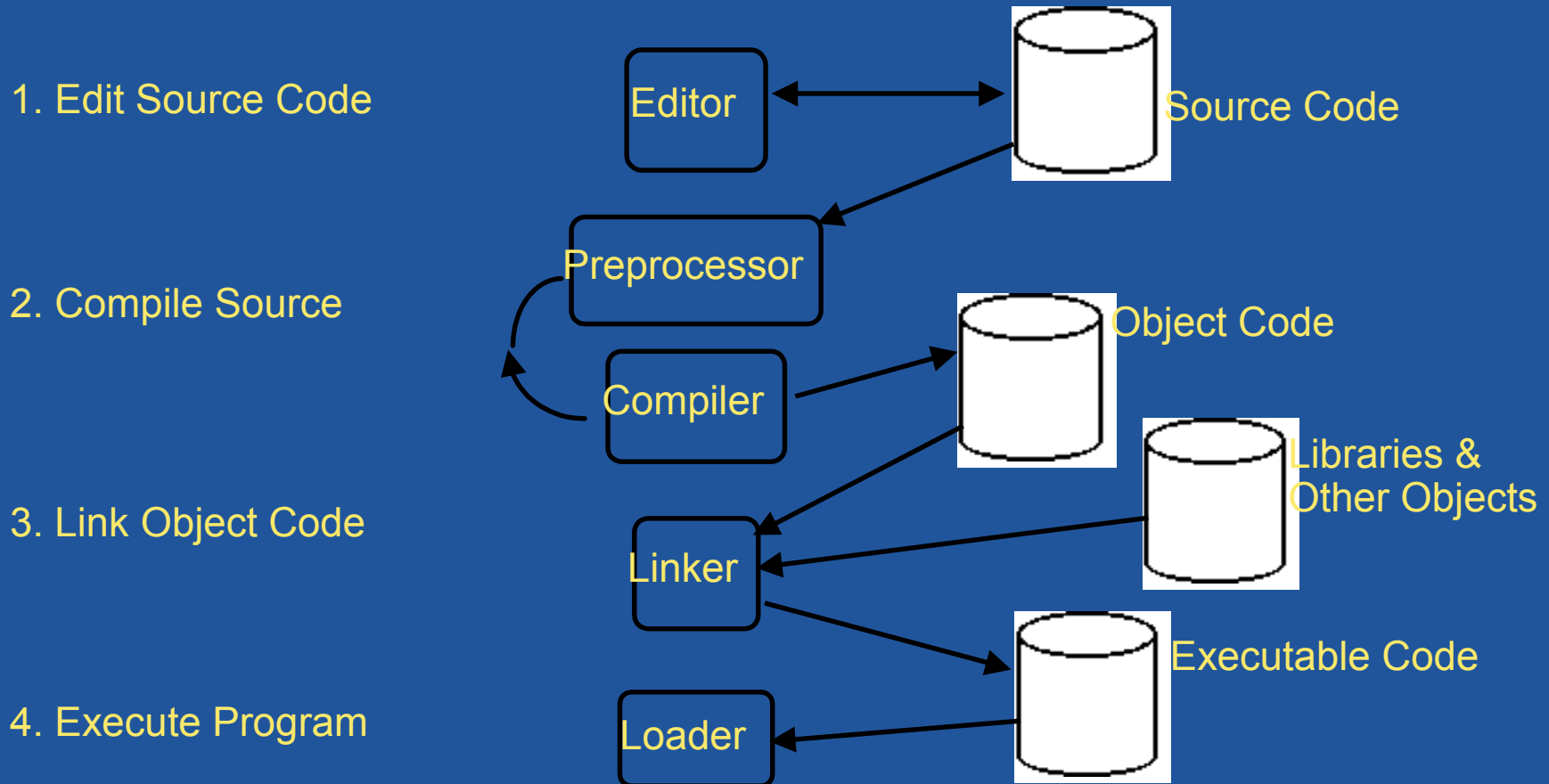
sw \$15, 4(\$2)

```
0000 1001 1100 0110 1010 1111 0101 1000
1010 1111 0101 1000 0000 1001 1100 0110
1100 0110 1010 1111 0101 1000 0000 1001
0101 1000 0000 1001 1100 0110 1010 1111
```

ALUOP[0:3] <= InstReg[9:11] & MASK

- o
- o

Passi per la costruzione di un programma



* these steps are repeated until your labs is working properly and you are ready to produce the final listing and output to turn in for a grade.

Concetti sulla memorizzazione

BIT - Binary Digit

- ❑ 8 bit per ogni byte

- ❑ Rappresenta una condizione binaria ON / OFF

BYTE – La più piccola unità di memoria indirizzabile;
tipicamente memorizza un carattere

WORD – Una unità di memoria indirizzabile su alcune macchine

Concetti sulla rappresentazione dell'informazione

Ogni numero, lettera, o carattere speciale è rappresentato, o codificato, come una combinazione di bit

- Ci sono due tabelle di codifica per i caratteri che sono diffuse:
 - **EBCDIC (IBM solamente)**
 - **ASCII**

Rappresentazione binaria, decimale, e ASCII

- **Un valore ASCII rappresenta sempre un carattere**
 - **alfanumerico** : 1, A, 5, c
 - **o speciale** : #, <FF>, ^C, ^L
- **Un valore decimale è in base 10; ...19 34 54 2001**
- **Un valore binario è in base 2; 100110101 0101 1101**
- **Un valore esadecimale è in base 16; usa 0-9, A-F**

Pattern di bit

I valori numerici sono memorizzati come numeri binari.

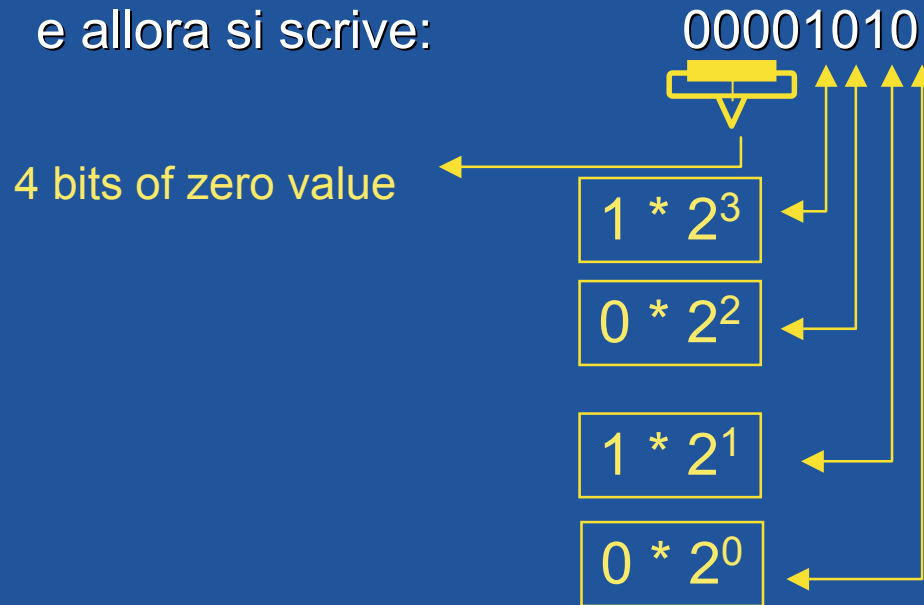
Il numero 10 rappresenta

$0 \times 10^0 + 1 \times 10^1$ in base 10 (decimale),

oppure

$0 \times 2^0 + 1 \times 2^1 + 0 \times 2^2 + 1 \times 2^3$ in base 2 (binaria)

e allora si scrive:



I caratteri sono numeri

Il numero decimale 10 è 00001010

il carattere '10' non è lo stesso che il valore decimale 10

- I caratteri sono DUE '1' e '0',
- memorizzati in DUE byte, rappresentante ognuno il codice ASCII relativo

	<u>Byte 1</u>	<u>Byte 2</u>
contenuto:	00110001	00110000
rappresenta:	'1' (decimal 49)	'0'(decimal 48)

Tutti i caratteri sono numeri

Il carattere '1' è memorizzato in un byte, come una sequenza di bit
Il carattere 'A' è memorizzato in un byte, come una sequenza di bit
Il numero 65 può essere memorizzato in un byte, come una sequenza di bit

Per un certo byte, la sequenza di bit ha

- Una rappresentazione binaria, come 00001010 or 01000001 (che è spesso espresso con il suo esadecimale equivalente; in questo caso, 0A and 41, rispettivamente)
- E un valore decimale, come 10 oppure 65
- E un valore ASCII

il programmatore sceglie quale valore usare!

.... Tutto dipende dal contesto!!!!

Alcuni codici ASCII

See ASCII character chart in Appendix D

How is the character 'A' represented?

- **as a decimal number: 65**
- **as a bit pattern: 01000001**
- **as a hexadecimal number: 41**

How is the character 'a' represented?

How is the ASCII character '1' stored differently from the number 1?

Codice ASCII

Byte	Cod.	Char	Byte	Cod.	Char	Byte	Cod.	Char	Byte	Cod.	Char
00000000	0	Null	00100000	32	Spc	01000000	64	@	01100000	96	`
00000001	1	Start of heading	00100001	33	!	01000001	65	A	01100001	97	a
00000010	2	Start of text	00100010	34	"	01000010	66	B	01100010	98	b
00000011	3	End of text	00100011	35	#	01000011	67	C	01100011	99	c
00000100	4	End of transmit	00100100	36	\$	01000100	68	D	01100100	100	d
00000101	5	Enquiry	00100101	37	%	01000101	69	E	01100101	101	e
00000110	6	Acknowledge	00100110	38	&	01000110	70	F	01100110	102	f
00000111	7	Audible bell	00100111	39	'	01000111	71	G	01100111	103	g
00001000	8	Backspace	00101000	40	(01001000	72	H	01101000	104	h
00001001	9	Horizontal tab	00101001	41)	01001001	73	I	01101001	105	i
00001010	10	Line feed	00101010	42	*	01001010	74	J	01101010	106	j
00001011	11	Vertical tab	00101011	43	+	01001011	75	K	01101011	107	k
00001100	12	Form Feed	00101100	44	,	01001100	76	L	01101100	108	l
00001101	13	Carriage return	00101101	45	-	01001101	77	M	01101101	109	m
00001110	14	Shift out	00101110	46	.	01001110	78	N	01101110	110	n
00001111	15	Shift in	00101111	47	/	01001111	79	O	01101111	111	o
00010000	16	Data link escape	00110000	48	0	01010000	80	P	01110000	112	p
00010001	17	Device control 1	00110001	49	1	01010001	81	Q	01110001	113	q
00010010	18	Device control 2	00110010	50	2	01010010	82	R	01110010	114	r
00010011	19	Device control 3	00110011	51	3	01010011	83	S	01110011	115	s
00010100	20	Device control 4	00110100	52	4	01010100	84	T	01110100	116	t
00010101	21	Neg. acknowledge	00110101	53	5	01010101	85	U	01110101	117	u
00010110	22	Synchronous idle	00110110	54	6	01010110	86	V	01110110	118	v
00010111	23	End trans. block	00110111	55	7	01010111	87	W	01110111	119	w
00011000	24	Cancel	00111000	56	8	01011000	88	X	01111000	120	x
00011001	25	End of medium	00111001	57	9	01011001	89	Y	01111001	121	y
00011010	26	Substitution	00111010	58	:	01011010	90	Z	01111010	122	z
00011011	27	Escape	00111011	59	;	01011011	91	[01111011	123	{
00011100	28	File separator	00111100	60	<	01011100	92	\	01111100	124	
00011101	29	Group separator	00111101	61	=	01011101	93]	01111101	125	}
00011110	30	Record Separator	00111110	62	>	01011110	94	^	01111110	126	~
00011111	31	Unit separator	00111111	63	?	01011111	95	_	01111111	127	Del

Codice ASCII esteso

Byte	Cod.	Char	Byte	Cod.	Char	Byte	Cod.	Char	Byte	Cod.	Char
10000000	128	Ç	10100000	160	á	11000000	192	+	11100000	224	Ô
10000001	129	ü	10100001	161	í	11000001	193	-	11100001	225	ß
10000010	130	é	10100010	162	ó	11000010	194	-	11100010	226	Û
10000011	131	â	10100011	163	ú	11000011	195	+	11100011	227	Ö
10000100	132	ä	10100100	164	ñ	11000100	196	-	11100100	228	ö
10000101	133	à	10100101	165	Ñ	11000101	197	+	11100101	229	Õ
10000110	134	â	10100110	166	ª	11000110	198	ä	11100110	230	µ
10000111	135	ç	10100111	167	•	11000111	199	Ã	11100111	231	þ
10001000	136	ê	10101000	168	¿	11001000	200	+	11101000	232	ð
10001001	137	ë	10101001	169	@	11001001	201	+	11101001	233	Ú
10001010	138	è	10101010	170	¬	11001010	202	-	11101010	234	Û
10001011	139	ï	10101011	171	½	11001011	203	-	11101011	235	Ü
10001100	140	î	10101100	172	¼	11001100	204	¡	11101100	236	ý
10001101	141	ï	10101101	173	í	11001101	205	-	11101101	237	ÿ
10001110	142	Ä	10101110	174	«	11001110	206	+	11101110	238	—
10001111	143	Å	10101111	175	»	11001111	207	©	11101111	239	•
10010000	144	Ê	10110000	176	-	11010000	208	ø	11110000	240	-
10010001	145	æ	10110001	177	-	11010001	209	Ð	11110001	241	±
10010010	146	Æ	10110010	178	-	11010010	210	Ê	11110010	242	-
10010011	147	ô	10110011	179	—	11010011	211	Ë	11110011	243	¾
10010100	148	õ	10110100	180	—	11010100	212	È	11110100	244	¶
10010101	149	ò	10110101	181	À	11010101	213	É	11110101	245	§
10010110	150	û	10110110	182	Á	11010110	214	Í	11110110	246	÷
10010111	151	ù	10110111	183	Â	11010111	215	Î	11110111	247	÷
10011000	152	ÿ	10111000	184	©	11011000	216	Ï	11111000	248	•
10011001	153	Û	10111001	185	—	11011001	217	+	11111001	249	•
10011010	154	Ü	10111010	186	—	11011010	218	+	11111010	250	•
10011011	155	ß	10111011	187	+	11011011	219	-	11111011	251	•
10011100	156	£	10111100	188	+	11011100	220	-	11111100	252	•
10011101	157	Ø	10111101	189	¢	11011101	221	—	11111101	253	•
10011110	158	×	10111110	190	¥	11011110	222	Ï	11111110	254	-
10011111	159	f	10111111	191	+	11011111	223	-	11111111	255	-

ASCII Completo

0		32		64	@	96	`	128	Ç	160	á	192	Ł	224	Ó
1	☉	33	!	65	A	97	a	129	ü	161	í	193	⊥	225	ß
2	☼	34	"	66	B	98	b	130	é	162	ó	194	⌊	226	Ô
3	♥	35	#	67	C	99	c	131	â	163	ú	195	⌋	227	Ò
4	♠	36	\$	68	D	100	d	132	ä	164	ñ	196	—	228	ô
5	♣	37	%	69	E	101	e	133	à	165	Ñ	197	⊕	229	Õ
6	♠	38	&	70	F	102	f	134	á	166	ª	198	ã	230	µ
7	·	39	'	71	G	103	g	135	ç	167	º	199	Ä	231	þ
8	▣	40	(72	H	104	h	136	ê	168	¿	200	⌌	232	Ɔ
9	○	41)	73	I	105	i	137	ë	169	®	201	⌍	233	Ú
10	⊠	42	*	74	J	106	j	138	è	170	¬	202	⌎	234	Û
11	♂	43	+	75	K	107	k	139	ï	171	½	203	⌏	235	Ü
12	♀	44	,	76	L	108	l	140	î	172	¼	204	⌐	236	Ý
13	♪	45	-	77	M	109	m	141	ì	173	¡	205	⌑	237	Ÿ
14	♫	46	.	78	N	110	n	142	Ë	174	«	206	⌒	238	ˉ
15	☼	47	/	79	O	111	o	143	Ä	175	»	207	◻	239	˘
16	▶	48	0	80	P	112	p	144	É	176	⦿	208	◊	240	-
17	◀	49	1	81	Q	113	q	145	æ	177	⦿	209	Ⓓ	241	±
18	↑	50	2	82	R	114	r	146	Æ	178	⦿	210	Ê	242	—
19	!!	51	3	83	S	115	s	147	ø	179		211	Ë	243	¾
20	¶	52	4	84	T	116	t	148	ö	180	⌋	212	È	244	¶
21	§	53	5	85	U	117	u	149	ò	181	Á	213	Ì	245	§
22	—	54	6	86	V	118	v	150	û	182	Â	214	Í	246	÷
23	↑	55	7	87	W	119	w	151	ù	183	Ã	215	Î	247	,
24	↑	56	8	88	X	120	x	152	ÿ	184	©	216	Ï	248	°
25	↓	57	9	89	Y	121	y	153	Ö	185	⌋	217	⌋	249	¨
26	→	58	:	90	Z	122	z	154	Ü	186	⌋	218	⌋	250	·
27	←	59	;	91	[123	{	155	ø	187	⌋	219	⌋	251	¹
28	⌌	60	<	92	\	124		156	£	188	⌋	220	⌋	252	³
29	↔	61	=	93]	125	}	157	∅	189	¢	221	⌋	253	²
30	▲	62	>	94	^	126	~	158	×	190	¥	222	⌋	254	■
31	▼	63	?	95	_	127	△	159	f	191	⌋	223	⌋	255	