# 40 – driver max7219 per matrice led 8x8 + bluetooth - max7219 driver, led matrix 8x8 & bluetooth (notes at end of this section)



Le matrici led 8x8 dotate di driver 7219 sono degli interessanti display da 64 led, con due fondamentali caratteristiche:

- necessitano di tre sole porte per poter essere pilotate
- possono essere collegate tra loro in cascata (non conosco il limite massimo, ma ho letto da qualche parte che esistono esempi con 8 moduli collegati in cascata)

In questo esercizio sono gestite tre matrici, sulle quali viene fatta scorrere una scritta digitata sulla tastiera di uno smartphone android collegato, via bluetooth ad Arduino. E' ovviamente possibile ripetere questo esercizio utilizzando un numero minore di matrici (anche solo una) o maggiore. Sara' in questi casi sufficiente inserire in maxDisplays (riga 28 del programma) il numero di matrici che si intendono usare (come dicevo, non sono certo che possa funzionare con piu' di 8 matrici).

E' anche possibile utilizzare la tastiera del monitor seriale o una tastiera PS2 al posto della tastiera bluetooth. In questo caso sara' sufficiente sostituire il prefisso "BT" di ogni istruzione "BT.available" e "BT.read" con, rispettivamente, "Serial" oppure "PS2keyboard" (in quest'ultimo caso bisognera' anche inserire la libreria "PS2keyboard.h").

Prima di compilare il programma e' necessario scaricare ed installare la libreria maxmatrix.h scaricabile da <u>https://code.google.com/archive/p/arudino-maxmatrix-library/downloads</u>

Per installare la libreria:

- Aprire il link
- Cliccare su maxmatrix.zip
- Selezionare maxmatrix
- Premere "estrai in"
- Inserire nel percorso di destinazione il percorso della cartella di download e premere ok
- Lanciare l'IDE di arduino
- Selezionare sketch->include library->add.zip library
- Selezionare la cartella di download->maxmatrix e premere "apri"
- Riavviare l'IDE e controllare che in sketch->include library sia presente la nuova libreria MaxMatrix

Se si utilizza la tastiera bluetooth di android bisogna anche connettere il proprio smartphone ad Arduino. Questi i passi:

- Si scarica da google play l'app *Bluetooth terminal* e la si installa sullo smartphone. *Bluetooth teminal* e' un programma che trasforma il telefono in un monitor seriale in grado di inviare, via bluetooth, le informazioni digitate sulla tastiera e di visualizzare eventuali messaggi ricevuti.
- Si connette ad arduino la scheda HC06, seguendo lo schema sottoriportato e subito il led di cui e' dotata lampeggera' velocemente, segno che e' attiva ma che non e' ancora stata associata ad un dispositivo bluetooth.
- Si attiva ora la connessione bluetooth sul telefono, (impostazioni -> bluetooth). Toccando bluetooth si apre, sempre sul telefono, la schermata con l'elenco dei dispositivi presenti in zona. Si seleziona HC-06 e si inserisce il pin 1234. Si attiva quindi, sempre sul telefono, l'app bluetooth terminal, si seleziona dal menu' in alto a destra la "connessione sicura" (o anche la connessione insicura, che funziona benissimo). Si stabilira' immediatamente la connessione, il led sulla scheda HC06 diventera' fisso ed il sistema sara' pronto ad operare.



**Nota:** Questo esercizio e questa nota sono parte di una serie che vede protagonisti Arduino ed alcuni dei componenti ad esso collegabili. Per la maggior parte degli esercizi e' anche disponibile un filmato su youtube.

- Esercizi facenti parte della raccolta
- Filmati presenti su youtube
- Informazioni su arduino e sui componenti collegabili (PDF scaricato nell'area di download)
- Breve manuale di programmazione (PDF scaricato nell'area di download)

Per eventuali chiarimenti o suggerimenti sul contenuto di questa scheda scrivere a <u>giocarduino@libero.it</u>

## Here some notes about this project, translated by google translator

 $\bigcirc$ 

The 8x8led matrix, equipped with 7219 driver, is an interesting 64led display, and has two basic characteristics:

- require only three pin to be managed
- can be connected in cascade (i do not know limit, but i read somewhere that there are examples with 8 modules connected)

In this exercise we manage three matrices, on which is slid a written, typed by a keyboard of an Android smartphone connected via bluetooth to Arduino. Obviously you can repeat this exercise using a smaller number of matrices (even only one) or greater. In this case will be simply enter in maxDisplays (line 28 on the program) the number of matrices you want use (as said, I'm not sure it will work with more than 8 matrices).

You can also use the keyboard of serial monitor or a PS2 keyboard instead of bluetooth keyboard. In this case will be simply replace the "BT" prefix of each instruction "BT.available" and "BT.read" with, respectively, "Serial" or "PS2keyboard" (in the latter case will be also must install the PS2keyboard.h library).

Before proceeding to program compilation must be installed, if not already done, the library:

 maxmatrix.h found <u>here</u> (this link will download a library which in turn contains other libraries. Once downloaded you have to open it and install only the maxmatrix.h library)

For library installation, see process shown in previous examples, and summarized in:

- library download in compressed form;
- Installation via IDE-> sketch-> includes Library-> add .zip library
- After installation please verify the library. It must be present in IDE-> sketch-> includes Library-> Contributed library

If you use the Android bluetooth keyboard, you must also connect your smartphone to Arduino.

These are the steps:

- download app "Bluetooth Terminal" from Google Play and install it on your smartphone. "Bluetooth teminal" is a program that turns the phone into a serial monitor, capable of sending, via bluetooth, the data typed on keyboard and display any received message.
- connect Arduino to HC06, following the schematic (see below) and immediately the hc06 led start blinking fast. That means that is active but not yet associated with a bluetooth device.
- Activate the bluetooth connection on your phone (Settings -> Bluetooth) and will appear list
  of active devices in the area. Select HC-06, and enter password 1234. Then launch, on
  phone, the "bluetooth terminal" app and select, from menu at the top right, the "secure
  connection" (or even the "insecure connection", which works fine). It will establish
  immediately a connection, the LED on HC06 will fixed and system will be ready to operate.

**Note:** This project and this note is part of a series that sees, as main characters, Arduino and some of connectable components. For most projects there is also a video on youtube.

- Projects collection
- Movies on youtube
- About Arduino and components (italian; pdf will be downloaded in your download area
- Quick programming guide (almost english; pdf will be downloaded in your download area)

For any questions or suggestions about this note (and on its english translation), please write to <u>giocarduino@libero.it</u> (simple words and short sentences, please)

## Materiali

- una (o piu') matrici led con driver 7219
- Una scheda bluetooth di tipo HC-06 (non obbligatoria)
- Un telefono android (non obbligatorio)
- Un po' di cavetteria

## Schema



https://code.google.com/p/arudino-maxmatrix-library/wiki/Example Display Scrolling Text

#### Arduino: matrice 8x8 con driver 7219 e bluetooth – bluetooth, 8x8 led matrix & max7219 driver

\* #include <SoftwareSerial.h> // libreria di gestione del sistema bluetooth
SoftwareSerial BT(6, 5); // porte alle quali devono essere collegati i pin TXD ed RXD della scheda bluetooth #include <MaxMatrix.h> // libreria di gestione del driver 7219 #include <avr/pgmspace.h> #define maxDisplays 3 // inserire il numero dei display 7219 che si intendono usare // insert number of 8x8 matrices you intend use const int lunghezzafrase = 50; // numero max di caratteri della frase che si vuole esporre sentence max lenght int contacaratteri = 0; // indice di scorrimento della tabella "stringa" in cui e' contenuta la frase da esporre int contatore = 0; // contatore di caratteri, utilizzato nella esposizione della frase byte Buf7219[7]; // "larghezza, altezza, dati[5]" - contenitore (zona di lavoro) di ogni singolo carattere da esporre. char string [lunghezzafrase]; // zona in cui deve essere inserita la frase da esporre const int data = 11; // DIN o MOSI da collegare alla porta 11 const int load = 10; // CS da collegare alla porta 10 const int clock = 13; // SCK (o clk) da collegare alla porta 13 11 MaxMatrix m(data, load, clock, maxDisplays); // parametri di configurazione delle connessioni delle matrici 8x8 /\* Tabella (qua' sotto) di trascodifica dei caratteri, caricata in memoria. Ogni riga di 7 caratteri definisce l'immagine di un carattere; l'indice di accesso ad ogni singolo elemento e' il valore ascii del carattere che si vuole esporre, diminuito di 32 (la tabella parte con lo spazio, che ha valore asci = 32) In ogni elemento sono presenti: larghezza, altezza e 5 byte, la cui configurazione binaria "disegna" il carattere da esporre. Per avere l'idea del disegno del carattere bisogna immaginare i cinque byte rappresentati in verticale: gli "1" corrispondono ai led che verranno accesi e fatti scorrere (il disegno del secondo elemento della tabella, corrispondente al "!", sembra abbastanza chiaro) Table (here below) to characters transcoding. Each row of 7 characters defines the image of a character; the index of access to every single item is the ascii value of the character that you want to expose, reduced by 32 (the table start with space, which has asci value = 32). In each element are present: width, height and 5 bytes, whose binary configuration "draws" the character to be exhibited. To get an idea must imagine the five bytes in vertical exibition: each "1" correspond to the LED that will be lit and made to slide (the second item in the table design, corresponding to "!", seems pretty clear) \* / PROGMEM const unsigned char CH[] = { 3, 8, B0000000, B0000000, B0000000, B0000000, B0000000, // spazio 1, 8, B1011111, B0000000, B0000000, B0000000, B0000000, // ! 3, 8, B0000011, B0000000, B0000011, B0000000, B0000000, // " 5, 8, B0010100, B0111110, B0010100, B0111110, B0010100, // # 4, 8, B0100100, B1101010, B0101011, B0010010, B0000000, // \$ 5, 8, B1100011, B0010011, B0001000, B1100100, B1100011, // 5, 8, B0110110, B1001001, B1010110, B0100000, B1010000, // & 1, 8, B0000011, B0000000, B0000000, B0000000, B0000000, // 3, 8, B0011100, B0100010, B1000001, B0000000, B0000000, // ( 3, 8, B1000001, B0100010, B0011100, B0000000, B0000000, // ) 5, 8, B0101000, B0011000, B0001110, B0011000, B0101000, // \* 5, 8, B0001000, B0001000, B0111110, B0001000, B0001000, // + 2, 8, B1100000, B0000000, B0000000, B0000000, B0000000, // , 4, 8, B0001000, B0001000, B0001000, B0001000, B0000000, // 2, 8, B1100000, B1100000, B0000000, B0000000, B0000000, // . 4, 8, B1100000, B0011000, B0000110, B0000001, B0000000, // / 4, 8, B0111110, B1000001, B1000001, B0111110, B0000000, // 0 3, 8, B1000010, B1111111, B1000000, B0000000, B0000000, // 1 4, 8, B1100010, B1010001, B1001001, B1000110, B0000000, // 2 4, 8, B0100010, B1000001, B1001001, B0110110, B0000000, // 3 4, 8, B0011000, B0010100, B0010010, B1111111, B0000000, // 4 4, 8, B0100111, B1000101, B1000101, B0111001, B0000000, // 5 4, 8, B0111110, B1001001, B1001001, B0110000, B0000000, // 6 4, 8, B1100001, B0010001, B0001001, B0000111, B0000000, // 7

4, 8, B0110110, B1001001, B1001001, B0110110, B0000000, // 8 4, 8, B0000110, B1001001, B1001001, B0111110, B0000000, // 9 2, 8, B0101000, B0000000, B0000000, B0000000, B0000000, // : 2, 8, B1000000, B0101000, B0000000, B0000000, B0000000, // ; 3, 8, B0010000, B0101000, B1000100, B0000000, B0000000, // <

3, 8	s,	B0010100,	в0010100,	в0010100,	в0000000,	в0000000,	11	=	
- / -	3,	B1000100,	в0101000,	в0010000,	в0000000,	в0000000,	11	>	
4, 8	3,	B0000010,	B1011001,	B0001001,	в0000110,	воооооо,	11	?	
5.8	3.	в0111110.	в1001001.	в1010101.	в1011101.	в0001110.	11	ß	
4.8	2	B1111110.	B0010001.	B0010001.	B1111110.	B0000000.	11	Δ	
1,0	, <b>'</b>	D1111110,	D10010001,	D0010001,	D1111110,	B00000000,	11	D	
4,0	, <b>,</b>	DIIIII,	D1001001,	D1001001,	B0110110,	воооооо,	·//	D	
4,8	· ·	BUIIIII,	BI000001,	BI000001,	B0100010,	воооооо,	11	-	
4, 8	<sup>,</sup>	BIIIIII,	BI000001,	BI000001,	B0111110,	вооооооо,	//	D	
4, 8	β,	B1111111,	B1001001,	B1001001,	B1000001,	воооооо,	//	Е	
4, 8	β,	B1111111,	B0001001,	B0001001,	B0000001,	воооооо,	//	F	
4, 8	3,	в0111110,	B1000001,	B1001001,	в1111010,	в0000000,	11	G	
4, 8	3,	B1111111,	B0001000,	B0001000,	B1111111,	в0000000,	11	Н	
3, 8	3.	B1000001,	B1111111,	B1000001,	в0000000,	в0000000,	11	I	
4.8	3.	в0110000.	в1000000.	в1000001.	в0111111.	в0000000.	11	J	
4.8	×.	B1111111	B0001000	B0010100.	B1100011	B0000000	11	ĸ	
1,0	, <b>'</b>	D1111111,	D0001000,	D0010100,	B1000011,	B00000000,	11	т	
4, 0 E 0	, <b>,</b>	DIIIIII,	B1000000,	D1000000,	B1000000,	Б0000000, Б1111111	·//	L M	
J, 0	· ·	DIIIIII,	B0000010,	B0001100,	B0000010,	DIIIIII,		141	
5,8	· ·	BIIIIII,	B0000100,	BUUUIUUU,	B0010000,	BIIIIII,	11	N	
4, 8	s,	B0111110,	BI000001,	BI000001,	B0111110,	воооооо,	//	0	
4, 8	3,	B1111111,	в0001001,	B0001001,	в0000110,	в0000000,	//	P	
4, 8	3,	B0111110,	B1000001,	B1000001,	B1011111,	воооооо,	//	Q	
4, 8	3,	B1111111,	B0001001,	B0001001,	в1110110,	в0000000,	11	R	
4, 8	3,	B1000110,	B1001001,	B1001001,	в0110010,	в0000000,	11	S	
5,8	3,	в0000001,	B0000001,	B1111111,	в0000001,	в0000001,	11	Т	
4.8	3.	в0111111.	в1000000.	в1000000.	в0111111.	в0000000.	11	U	
5.8	}.	B0001111.	B0110000.	B1000000.	B0110000.	B0001111.	11	V	
5,8	2	B0111111	B1000000	B0111000	B1000000	B0111111	11	TAT	
5,0	, <b>,</b>	D0111111,	D1000000,	D0111000,	D1000000,	DUIIIII,	<i>''</i>	v	
J, 0	· ·	BII00011,	B0010100,	BUUUIUUU,	B0010100,	BII00011,		A 17	
5,8	· ·	BUUUUIII,	BUUUIUUU,	BIII0000,	B0001000,	B0000111,	11	ĭ	
4,8	s,	BII00001,	BI010001,	BI001001,	BI000111,	воооооо,	11	Z	
2,8	3,	B1111111,	B1000001,	в0000000,	воооооо,	в0000000,	//	[	
4, 8	β,	B0000001,	в0000110,	в0011000,	в1100000,	воооооо,	//	barra	inversa
2,8	β,	B1000001,	B1111111,	воооооо,	воооооо,	в0000000,	//	]	
3, 8	3,	B0000010,	в0000001,	B0000010,	в0000000,	в0000000,	11	hat	
4, 8	3,	B1000000,	в1000000,	B1000000,	в1000000,	в0000000,	11		
2, 8	3.	в0000001,	в0000010,	в0000000,	в0000000,	в0000000,	11	<u>`</u>	
4.8	×.	B0100000.	B1010100.	B1010100.	B1111000.	в0000000.	11	a	
4 8	2	B1111111	B1000100	B1000100,	B01110000,	B00000000,	11	h	
4,0	, <b>,</b>	D1111111,	B1000100,	D1000100,	B01110000,	B00000000,	11	0	
4,0	, <b>,</b>	DOIII000,	D1000100,	D1000100,	Б0000000, р1111111	воооооо,		ر م	
4,8	<b>`</b>	BUIII000,	B1000100,	B1000100,	BIIIIII,	воооооо,	- / /	a	
4,8	· ·	B0111000,	BI010100,	BI010100,	B0011000,	воооооо,	11	e	
3,8	3,	B0000100,	B1111110,	B0000101,	воооооо,	воооооо,	11	f	
4, 8	<sup>3</sup> ,	B1001100,	B1010010,	B1010010,	в0111100,	в0000000,	//	g	
4, 8	β,	B1111111,	в0000100,	в0000100,	в1111000,	в0000000,	//	h	
3,8	β,	B1000100,	B1111101,	B1000000,	воооооо,	воооооо,	//	i	
4, 8	3,	B1000000,	B1000000,	B1000000,	в0111101,	в0000000,	11	j	
4, 8	3,	B1111111,	B0010000.	в0101000,	B1000100,	в0000000,	11	k	
3 0					B0000000	воооооо.	11	1	
J, U	β,	B1000001,	B1111111,	B1000000,	D0000000,	200000000		<u> </u>	
5,8	3, 3.	B1000001, B1111100,	B1111111, B0000100,	B1000000, B1111100,	B0000100,	B1111000.	11	m	
5,8 4 8	3, 3,	B1000001, B1111100,	B1111111, B0000100,	B1000000, B1111100, B0000100	B0000100, B1111000	B1111000,	11	m	
5,8 4,8	<sup>3</sup> , <sup>3</sup> ,	B1000001, B1111100, B1111100,	B111111, B0000100, B0000100, B1000100	B1000000, B1111100, B0000100,	B0000100, B1111000,	B1111000, B0000000,		m n	
5, 8 4, 8 4, 8	3, 3, 3,	B1000001, B1111100, B1111100, B0111000,	B1111111, B0000100, B0000100, B1000100,	B1000000, B1111100, B0000100, B1000100,	B0000100, B1111000, B0111000,	B1111000, B0000000, B0000000,	    	m n o	
5, 8 4, 8 4, 8 4, 8	<sup>3</sup> , <sup>3</sup> , <sup>3</sup> ,	B1000001, B1111100, B1111100, B0111000, B1111100,	B1111111, B0000100, B0000100, B1000100, B0100100,	B1000000, B1111100, B0000100, B1000100, B0100100,	B0000100, B1111000, B0111000, B0011000,	B1111000, B0000000, B0000000, B0000000,	· · · · · · · · · · · · · · · · · · ·	m n o p	
5, 8 5, 8 4, 8 4, 8 4, 8 4, 8	<sup>3</sup> , <sup>3</sup> , <sup>3</sup> , <sup>3</sup> , <sup>3</sup> ,	B1000001, B111100, B111100, B0111000, B111100, B0011000,	B111111, B0000100, B0000100, B1000100, B0100100, B0100100,	B1000000, B1111100, B0000100, B1000100, B0100100, B0100100,	B0000100, B1111000, B0111000, B0011000, B111100,	B1111000, B0000000, B0000000, B0000000, B0000000,	   	m n o p q	
5, 8 5, 8 4, 8 4, 8 4, 8 4, 8 4, 8	<sup>3</sup> , <sup>3</sup> , <sup>3</sup> , <sup>3</sup> , <sup>3</sup> ,	B1000001, B111100, B111100, B0111000, B1111100, B0011000, B1111100,	B111111, B0000100, B0000100, B1000100, B0100100, B0100100, B0001000,	B1000000, B1111100, B0000100, B1000100, B0100100, B0100100, B0000100,	B0000100, B1111000, B0111000, B0011000, B1111100, B0000100,	B1111000, B0000000, B0000000, B0000000, B0000000, B0000000,	          	m n o p q r	
5, 8 5, 8 4, 8 4, 8 4, 8 4, 8 4, 8 4, 8	<sup>3</sup> , <sup>3</sup> , <sup>3</sup> , <sup>3</sup> , <sup>3</sup> , <sup>3</sup> ,	B1000001, B1111100, B1111100, B0111000, B1111100, B0011000, B1111100, B1001000,	B1111111, B0000100, B1000100, B1000100, B0100100, B0100100, B1010100,	B1000000, B1111100, B0000100, B1000100, B0100100, B0100100, B1010100,	B0000100, B1111000, B0111000, B0011000, B1111100, B0000100, B0100100,	B1111000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000,	          	m n o p q r s	
5, 8 4, 8 4, 8 4, 8 4, 8 4, 8 4, 8 4, 8 3, 8	<sup>3</sup> , <sup>3</sup> , <sup>3</sup> , <sup>3</sup> , <sup>3</sup> , <sup>3</sup> , <sup>3</sup> , <sup>3</sup> ,	B1000001, B1111100, B1111100, B0111000, B1111100, B0011000, B1111100, B1001000, B00001000,	B1111111, B0000100, B1000100, B0100100, B0100100, B0100100, B1010100, B0111111,	B1000000, B111100, B0000100, B1000100, B0100100, B0100100, B0000100, B1010100, B1000100,	B0000100, B1111000, B0111000, B011000, B1111100, B0000100, B0100100, B0000000,	B1111000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000,	          	m n o p q r s t	
5, 8 5, 8 4, 8 4, 8 4, 8 4, 8 4, 8 4, 8 3, 8 4, 8		B1000001, B1111100, B1111100, B111100, B111100, B0011000, B1111100, B1001000, B0000100, B0000100, B0111100,	B111111, B0000100, B0000100, B1000100, B0100100, B0001000, B1010100, B0111111, B1000000,	B1000000, B111100, B0000100, B1000100, B0100100, B0000100, B1010100, B1000100, B1000100,	B0000100, B1111000, B0111000, B0011000, B1111100, B0000100, B0100100, B1111100,	B1111000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000,	· · · · · · · · · · · · · · · · · · ·	m n o p q r s t u	
5, 8 5, 8 4, 8 4, 8 4, 8 4, 8 4, 8 4, 8 4, 8 4		B1000001, B1111100, B1111100, B111100, B111100, B0011000, B1010000, B0001000, B000100, B011100, B0011100,	B111111, B0000100, B0000100, B1000100, B0100100, B0100100, B1010100, B0111111, B1000000, B0100000,	B1000000, B111100, B0000100, B100100, B0100100, B0000100, B1010100, B1000100, B1000000, B1000000,	B0000100, B1111000, B0111000, B1111000, B1111100, B0000100, B0100100, B1111100, B1111100,	B1111000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0011100,	· · · · · · · · · · · · · · · · · · ·	m n o p q r s t u v	
5, 8 5, 8 4, 8 4, 8 4, 8 4, 8 4, 8 3, 8 4, 8 5, 8 5, 8		B1000001, B1111100, B1111100, B0111000, B1111100, B0011000, B1001000, B0001000, B0111100, B0011100, B0011100,	B111111, B0000100, B0000100, B1000100, B0100100, B0100100, B0101000, B0111111, B1000000, B0100000, B1000000,	B1000000, B111100, B0000100, B1000100, B0100100, B0100100, B1010100, B1000100, B1000000, B1000000, B0111100,	B0000100, B1111000, B0111000, B0011000, B1111100, B0000100, B0100100, B1111100, B0100000, B1000000,	B1111000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0011100, B011100,		m nopq r s t u v w	
5, 8 5, 8 4, 8 4, 8 4, 8 4, 8 4, 8 5, 8 5, 8 5, 8		B1000001, B1111100, B1111100, B111100, B111100, B111100, B111100, B111100, B0001000, B011100, B0111100, B1100100.	B111111, B0000100, B0000100, B1000100, B0100100, B0001000, B1010100, B0111111, B1000000, B0100000, B0100000,	B1000000, B111100, B0100100, B0100100, B0100100, B0000100, B1010100, B1000100, B1000000, B1000000, B0111100, B0010000.	B0000100, B1111000, B0111000, B0011000, B1111100, B0000100, B0100100, B111100, B111100, B1000000, B1000000, B0101000.	B1111000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0011100, B1000100.		m nop q r s t u v w x	
5, 8 5, 8 4, 8 4, 8 4, 8 4, 8 4, 8 4, 8 4, 8 4		B1000001, B1111100, B1111100, B1111100, B1111100, B1011000, B1111100, B1001000, B001100, B0111100, B0011100, B1000100, B10001110	B1111111, B0000100, B0000100, B0100100, B0100100, B0100100, B0101000, B0111111, B1000000, B0100000, B0100000, B01010000, B10100000	B1000000, B111100, B0000100, B0100100, B0100100, B0000100, B1000100, B1000100, B1000000, B1000000, B011100, B0010000	B0000100, B1111000, B011000, B011000, B111100, B0000100, B0100100, B1100000, B111100, B0100000, B1000000, B01010000, B0111110	B1111000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0011100, B10100, B1000100,		m n o p q r s t u v w x x	
5, 8 5, 8 4, 8 4, 8 4, 8 4, 8 4, 8 4, 8 4, 8 4		B1000001, B1111100, B1111100, B1111100, B011000, B111100, B1011000, B101100, B001100, B0111100, B0111100, B1001100, B10001100, B11000100,	B111111, B0000100, B0000100, B0100100, B0100100, B0001000, B1010100, B0111111, B1000000, B0100000, B1000000, B1010000, B1010000, B1010000,	B1000000, B111100, B000100, B100100, B0100100, B000100, B1000100, B1000100, B1000000, B1000000, B0111100, B0010000, B1010000, B1010000,	B0000100, B1111000, B0111000, B011000, B111100, B0000100, B0100100, B111100, B0100000, B1000000, B1000000, B0101000, B0111110,	B1111000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0011100, B0111100, B1000100, B0000000, B0000000	                	m n o p q r s t u v w x y z	
5, 8 5, 8 4, 8 4, 8 4, 8 4, 8 4, 8 5, 8 5, 8 5, 8 4, 8 5, 8 4, 8 5, 8 4, 8 5, 8 4, 8 5, 9 5, 9 5		B1000001, B1111100, B1111100, B111100, B111100, B1011000, B1011000, B000100, B0011100, B0111100, B0111100, B1001100, B1001110, B1001100,	B111111, B0000100, B0000100, B1000100, B0100100, B0100100, B0101000, B1010100, B1010000, B1000000, B1000000, B1010000, B1010000, B1010100, B0110110,	B1000000, B111100, B0000100, B100100, B0100100, B0000100, B1010100, B1000100, B1000000, B1000000, B0111100, B0010000, B1010000, B10001000,	B0000100, B1111000, B0111000, B0111000, B111100, B0000100, B0100100, B110100, B111100, B1000000, B1100000, B0101000, B011110, B0000000,	B1111000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0011100, B1001100, B1000100, B0000000, B0000000, B0000000,	                   	m n o p q r s t u v w x y z	
5,       8         5,       8         4,       8         4,       8         4,       8         4,       8         5,       8         4,       8         5,       8         4,       8         5,       8         5,       8         5,       8         5,       8         3,       8         3,       8		B1000001, B1111100, B1111100, B111100, B0011000, B101000, B1001000, B000100, B0111100, B0011100, B1001100, B1001100, B1001110, B100100, B1001000, B0001000,	B111111, B0000100, B000100, B1000100, B0100100, B0100100, B0101000, B0101000, B0100000, B0100000, B0101000, B1010000, B0101000, B0101100, B0101100, B0101110,	B1000000, B1111100, B0000100, B0100100, B0100100, B0000100, B1000100, B1000100, B1000000, B011100, B0010000, B1010000, B1001100, B1001100, B1000001,	B0000100, B1111000, B0111000, B0111000, B111100, B0000100, B0100100, B1111100, B0100000, B1111100, B0100000, B0101000, B0111110, B0000000, B0000000, B0000000, B0000000,	B1111000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B011100, B0111100, B1000100, B0000000, B0000000, B0000000,	                   	m n o p q r s t u v w x y z {	
5, 8 5, 8 4, 8 4, 8 4, 8 4, 8 4, 8 5, 8 5, 8 5, 8 3, 8 1, 8 5, 8 4, 8 5, 8 5		B1000001, B1111100, B1111100, B0111000, B1111100, B0011000, B1001000, B0111100, B0011100, B0011100, B1001100, B1001110, B1001100, B1001000, B1111111,	B111111, B0000100, B0000100, B1000100, B0100100, B0100100, B0101000, B0101000, B0100000, B1000000, B1010000, B1010000, B1010100, B0110110, B0000000,	B1000000, B111100, B0000100, B0100100, B0100100, B0000100, B1000100, B1000100, B1000000, B0111100, B0010000, B1001000, B1001000, B1000001, B0000000,	B0000100, B0111000, B0111000, B0111000, B0100100, B0100100, B0100100, B0100000, B1000000, B0101000, B0101000, B0111110, B0000000, B0000000, B0000000, B0000000,	B1111000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0011100, B011100, B1000100, B0000000, B0000000, B0000000, B0000000,	                      	mnopqrstuvwxyz{	
5, 8 5, 8 4, 8 4, 8 4, 8 4, 8 4, 8 4, 8 5, 8 8 5, 8 8 5, 8 8 3, 8 1, 8 3, 8 1, 8 3, 8 1,		B1000001, B1111100, B1111100, B1111100, B1111100, B1011000, B1111100, B1001000, B001100, B0011100, B1001100, B1001100, B1001110, B1100100, B111111, B1000001,	B111111, B0000100, B0000100, B1000100, B0100100, B0100100, B0101000, B0101000, B1010000, B1010000, B1010000, B1010000, B1010100, B0110110, B0110110,	B1000000, B111100, B0000100, B0100100, B0100100, B0000100, B1000100, B1000000, B1000000, B1010000, B1010000, B1001000, B1001100, B1000001, B0000000, B0001000,	B0000100, B1111000, B0111000, B0111000, B1111100, B0000100, B0100100, B0100000, B111100, B0100000, B0101000, B0101000, B0101000, B0000000, B0000000, B0000000,	B1111000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B011100, B1000100, B0000000, B0000000, B0000000, B0000000, B0000000,	                      	mnopqrstuvwxyz{  }	
5, 8 5, 8 4, 8 4, 8 4, 8 4, 8 4, 8 4, 8 5, 8 5, 8 5, 8 5, 8 3, 8 1, 8 3, 8 4, 8 3, 8 4, 8 5, 8 5, 8 5, 8 5, 8 5, 8 5, 8 5, 8 5		B1000001, B1111100, B1111100, B0111000, B1111100, B1011000, B1011000, B1001000, B0011100, B0011100, B1001100, B1001110, B1001100, B1001100, B1111111, B1000001, B0001000,	B111111, B0000100, B0000100, B0100100, B0100100, B0001000, B0101000, B0111111, B1000000, B0100000, B1010000, B1010000, B1010100, B0110110, B0000000, B0110110, B0000100,	B1000000, B111100, B0000100, B0100100, B0100100, B0000100, B1000100, B1000100, B1000000, B1000000, B1010000, B1010000, B1001100, B1000001, B0001000, B0001000,	B0000100, B1111000, B011000, B011000, B111100, B0000100, B0100100, B1100000, B111100, B0100000, B0101000, B011110, B0000000, B0000000, B0000000, B0000000, B0000000,	B1111000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0011100, B0111100, B1001000, B0000000, B0000000, B0000000, B0000000, B0000000,	· · · · · · · · · · · · · · · · · · ·	mnopqrstuvwxyz{  }~	
5, 8 5, 8 4, 8 4, 8 4, 8 4, 8 4, 8 3, 8 5, 8 5, 8 5, 8 5, 8 3, 8 3, 8 3, 8 4, 8 3, 8 3, 8 3, 8 3, 8 3, 8 3, 8 3, 8 3		B1000001, B1111100, B1111100, B1111100, B011000, B111100, B1011000, B0001000, B0111100, B0011100, B1001100, B1001100, B1001100, B1100100, B1111111, B1000001, B0001000,	B111111, B0000100, B0000100, B1000100, B0100100, B0001000, B0101000, B0101000, B1000000, B1000000, B1010000, B1010000, B1010100, B0110110, B0000000, B0110110, B0000100,	B1000000, B111100, B0000100, B0100100, B0100100, B0000100, B1000100, B1000100, B1000000, B1000000, B1010000, B1010000, B1001100, B1000001, B0000000, B0001000, B0001000,	B0000100, B1111000, B0111000, B0111000, B111100, B0000100, B0100100, B0100000, B1111100, B0100000, B0101000, B0101000, B0111110, B0000000, B0000000, B0000000, B0000000, B0000100,	B1111000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0011100, B0111100, B1000100, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000,	· · · · · · · · · · · · · · · · · · ·	mnopqrstuvwxyz{  }~	
5, 8 5, 8 4, 8 4, 8 4, 8 4, 8 4, 8 4, 8 5, 8 5, 8 4, 8 5, 8 4, 8 5, 8 1, 8 3, 8 1, 8 3, 8 }; //		B1000001, B1111100, B1111100, B1111100, B0011000, B1011000, B1001000, B000100, B0011100, B0011100, B1001100, B1001100, B1001100, B1001000, B1111111, B1000001, B0001000,	B111111, B0000100, B0000100, B1000100, B0100100, B0100100, B0101000, B0101000, B0100000, B1000000, B1010000, B1010000, B1010100, B0100100, B0110110, B0000100,	B1000000, B111100, B0000100, B100100, B0100100, B0000100, B1000100, B1000100, B1000000, B1000000, B1010000, B1010000, B1001000, B1000000, B0001000, B0001000,	B0000100, B0111000, B0111000, B0111000, B111100, B0000100, B0100100, B0100000, B1111100, B0100000, B0101000, B0101000, B0101000, B0000000, B0000000, B0000000, B0000000, B0000100,	B1111000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0011100, B0011100, B1000100, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000,	· · · · · · · · · · · · · · · · · · ·	mnopqrstuvwxyz{  }~	
5, 8 5, 8 4, 8 4, 8 4, 8 4, 8 4, 8 4, 8 5, 8 5, 8 4, 8 5, 8 4, 8 5, 8 4, 8 5, 8 4, 8 5, 8 4, 8 5, 8 5, 8 6, 8 5, 8 6, 8 7, 8 7		B1000001, B1111100, B1111100, B1111100, B0011000, B1011000, B1001000, B000100, B0011100, B0011100, B1001100, B1001100, B1001100, B1001000, B1111111, B1000001, B1001000,	B111111, B0000100, B000100, B1000100, B0100100, B0100100, B0100100, B0101000, B1010000, B0100000, B1010000, B1010000, B0101000, B0110110, B0110110, B0100100, B0110110, B0100100, B0110110, B0000100,	B1000000, B111100, B0000100, B0100100, B0100100, B0000100, B1000100, B1000100, B1000000, B1000000, B1010000, B1010000, B1000001, B0000000, B0001000, B0001000, B0001000, B0001000, B0001000, B0001000, B0001000,	B0000100, B0111000, B0111000, B0111000, B011000, B1111100, B0100100, B0100100, B1111100, B0100000, B0101000, B0101000, B0101000, B0000000, B0000000, B0000000, B0000000, B0000100, B0000100,	B1111000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0011100, B0111100, B0111100, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000,	// // // // // // // // // // // //	m n o p q r s t u v w x y z {   } ~ caratte	ere ************
5, 8 4, 8 4, 8 4, 8 4, 8 4, 8 4, 8 5, 8 4, 8 5, 8 4, 8 5, 8 4, 8 3, 8 4, 8 3, 8 4, 8 5, 8 6, 8 6, 8 6, 8 6, 8 6, 8 6, 8 6, 8 7, 8 8, 8 7, 8 8, 8 7, 8 8, 8 7, 8 8, 8		B1000001, B1111100, B1111100, B111100, B0011000, B1011000, B1001000, B0001000, B0111100, B0011100, B1001100, B1001100, B1001100, B1001100, B1111111, B1000001, B1111111, B10001000,	B111111, B0000100, B000100, B1000100, B0100100, B0100100, B0100100, B0101000, B0100000, B0100000, B0100000, B0101000, B0101000, B0110110, B0000000, B0110110, B0000100, B0110110, B0000100,	B1000000, B111100, B0000100, B0100100, B0100100, B0000100, B1000100, B1000100, B1000000, B1000000, B1010000, B1010000, B1000001, B0001000, B000100, B000000, B000000, B000000, B000000, B000000, B00000, B00000, B000000, B000000, B000000, B000000, B000000, B00000, B00000, B000000, B000000, B000000, B000000, B000000, B00000, B00000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B00000, B00000, B00000, B000000, B000000, B000000, B00000, B00000, B00000, B00000, B00000, B0000, B000, B0000, B0000, B000, B000, B000, B000, B000, B000, B000, B000, B000, B000, B00, B000, B00, B000, B0,	B0000100, B0111000, B0111000, B0111000, B011000, B0100100, B0100100, B0100000, B1111100, B0100000, B0101000, B0101000, B0000000, B0000000, B0000000, B0000000, B0000100, B0000100, B0000100, B0000100, B0000100, B0000100, B0000100, B0000100, B0000100, B0000100, B0000100, B0000000, B0000100, B000000, B0000000, B00000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B00000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B00000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B00000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B00000, B00000, B00000, B00000, B0000, B0000, B0000, B0000, B000, B000, B000, B000, B000, B000, B000, B000, B000, B0, B	B1111000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0111100, B0111100, B0111100, B000000, B000000, B000000, B000000, B000000, B00000, B00000, B00000, B00000, B00000, B0000, B0000, B0000, B0000, B0000, B0000, B000, B000, B000, B000, B000, B000, B000, B000, B	//////////////////////////////////////	m n o p q r s t u v w x y z {   } ~ caratte	976 ************************************
5, 8 5, 8 4, 8 4, 8 4, 8 4, 8 4, 8 4, 8 5, 8 5, 8 4, 8 5, 8 3, 8 3, 8 1, 8 8 4, 8 1, 8 4, 8 1, 8 4, 8 5, 8 1,		B1000001, B1111100, B1111100, B1111100, B0011000, B1111100, B1001000, B0001000, B0011100, B0011100, B1001100, B1001100, B1001100, B1001000, B1111111, B1000001, B0001000,	B111111, B0000100, B000100, B1000100, B0100100, B0100100, B0101000, B0101000, B0101000, B0100000, B1010000, B1010000, B0101000, B0110110, B0000000, B0110110, B0000100, B0110110, B0000100,	B1000000, B111100, B0000100, B0100100, B0100100, B0100100, B1000100, B1000100, B1000000, B1000000, B1010000, B1001000, B1000000, B000100, B0000100, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B00000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B00000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B00000, B00000, B00000, B00000, B0000, B000, B0000, B0000, B000, B000, B000, B000, B000, B000, B000, B000, B000, B000, B000, B000, B00, B000, B000, B00, B000, B0, B	B0000100, B0111000, B0111000, B0111000, B0100100, B0100100, B0100100, B0100000, B1111100, B0100000, B0101000, B0101000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000100, B0000100, B0000100, B0000100, B0000100, B0000100, B0000100, B0000100, B0000100, B0000100, B0000100, B0000000, B0000000, B0000000, B0000100, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B000000, B000000, B000000, B00000, B000000, B000000, B00000, B000000, B000000, B0000000, B000000, B000000, B000000, B000000, B000000, B00000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B00000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B00000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B00000, B00000, B000000, B000000, B000000, B000000, B00000, B00000, B0000, B0000, B0000, B0000, B0000, B000, B000, B0000, B000, B000, B000, B00, B000, B0	B1111000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0011100, B011100, B1000100, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B000000, B00000, B00000, B00000, B0000, B0000, B0000, B000, B000, B000, B000, B000, B000, B000, B000, B000, B000, B00, B000, B00, B00, B00, B00, B00, B00, B00, B00, B00, B00, B00, B00, B00, B00, B00, B0, B	//////////////////////////////////////	m n o p q r s t u v w x y z {   } ~ caratté *******	97e ************************************
5, 8 5, 8 4, 8 4, 8 4, 8 4, 8 4, 8 4, 8 5, 8 5, 8 5, 8 5, 8 5, 8 3, 8 3, 8 1, 8 3, 8 4, 8 5, 8 5, 8 5, 8 5, 8 5, 8 5, 8 5, 8 5	· · · · · · · · · · · · · · · · · · ·	B1000001, B1111100, B1111100, B1111100, B0111000, B1111100, B1001000, B1001000, B0001100, B0011100, B1001100, B1001110, B1001100, B1001100, B1001100, B1001000, B1111111, B1000001, B0001000, B1111111, B1000001, B1001000, B1111111, B1000001, B1001000, B1111111, B1000001, B1001000, B1111100, B1001000, B111100, B1001000, B111100, B1001000, B111100, B100000, B111100, B100000, B111100, B100000, B100100, B100000, B10000, B10000, B10000, B10000, B100000, B100000, B10000, B10000, B100000, B1000, B10000, B10000, B10000, B10000, B1000, B10000, B100, B1000, B1000, B100, B1000, B1000, B100, B100, B10, B1	Billilli, Bolooloo, Boliolio, Bolooloo, Bolio, Bolio, Bolio, Bolio, Bolio, Bolio, Bolio, Bolio, Bolio, Bol	B1000000, B111100, B0000100, B100100, B0100100, B0000100, B1000100, B1000100, B1000000, B1000000, B1010000, B1010000, B1010000, B1001000, B000100, B0000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B00000, B0000, B00000, B0000, B000, B000, B000, B000, B000, B000, B000, B000, B000, B	B0000100, B0111000, B0111000, B0111000, B0100100, B0100100, B0100100, B0100000, B1111100, B0100000, B0100000, B0100000, B0000000, B0000000, B0000000, B0000000, B0000100, B0000100, B0000100, B0000100, B0000100, B0000100, B0000100, B0000100, B0000100, B0000100, B0000100, B0000000, B0000100, B000000, B0000000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B000000, B000000, B000000, B00000, B000000, B000000, B00000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B00000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B00000, B00000, B000000, B000000, B000000, B00000, B00000, B0000, B0000, B0000, B0000, B0000, B000, B000, B000, B000, B000, B000, B00, B000, B0	B1111000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0011100, B0111100, B1001000, B000000, B000000, B000000, B000000, B000000, B00000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B0000, B00000, B00000, B0000, B000, B000, B000, B000, B000, B000, B000, B00,	//////////////////////////////////////	m n o p q r s t u v w x y z {   } ~ caratte *******	ere ***********************************
<pre>5, 8 5, 8 4, 8 4, 8 4, 8 4, 8 4, 8 4, 8 3, 8 5, 8 5, 8 5, 8 5, 8 3, 8 3, 8 3, 8 3, 8 3, 8 3, 8 3, 8 4, 8 }; // ***** // **** // **** // **** // **** // **** // **** // ***** // ***** // ***** // ****** // ******* // ********</pre>	<pre> } , , , , , , , , , , , , , , , , , ,</pre>	B1000001, B1111100, B1111100, B1111100, B011000, B111100, B1001000, B1001000, B0001000, B0011100, B1001100, B1001100, B1001100, B1001000, B1111111, B1000001, B0001000, B1111111, B1000001, B0001000, B1111111, B1000001, B0001000, B1111111, B1000001, B1111111, B1000001, B11111111, B1000001, B1111100, B1111100, B1111100, B1111100, B1111100, B1111100, B1111100, B1111100, B1111100, B1111100, B100000, B1111100, B1111100, B1111100, B100000, B1111100, B100100, B1111100, B100100, B1111100, B100000, B100100, B1001100, B100100, B100100, B100100, B100100, B100100, B100100, B1001100, B100100, B1001100, B100100, B1001100, B10000, B1001100, B100000, B10000, B10000, B10000, B10000, B10000, B10000, B10000, B1000, B10000, B10000, B10000, B10000, B1000, B10000, B10000, B1000, B10000, B10000, B100, B1000, B1000, B100, B100, B100, B100, B100, B100, B100, B100, B10, B1	Billilli, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolio, Bolio, Bolio, Bolio, Bolio, Bolio, Bolio, Bolio, Bolio, Bol	B1000000, B111100, B0000100, B0100100, B0100100, B0000100, B1000100, B1000000, B1000000, B1000000, B1010000, B1010000, B1001000, B000100, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B00000, B00000, B00000, B00000, B00000, B0000, B0000, B0000, B000, B000, B000, B000, B000, B000, B000, B000, B00,	B0000100, B001000, B1111000, B011000, B111100, B0000100, B0100100, B0100000, B111100, B0100000, B1000000, B0101000, B0101000, B0000000, B0000000, B0000000, B0000000, B0000100, B0000100, B0000100, B0000100, B0000100, B0000100, B0000100, B0000100, B0000100, B000000, B0000000, B00000, B0000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B00000, B00000, B00000, B000000, B00000, B00000, B00000, B0000, B000, B0000, B0000, B000, B000, B000, B000, B000, B000, B000, B000, B000, B000, B00, B000, B	B1111000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0011100, B0111100, B1001000, B000000, B000000, B000000, B000000, B000000, B00000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B0000, B00000, B00000, B0000, B000, B0000, B000, B000, B000, B000, B000, B000, B00, B000, B	//////////////////////////////////////	m n o p q r s t u v w x y z { {   } } ~ caratte	ere ***********************************
<pre>5, 8 5, 8 4, 8 4, 8 4, 8 4, 8 4, 8 4, 8 3, 8 5, 8 5, 8 5, 8 5, 8 3, 8 3, 8 3, 8 3, 8 4, 8 }; // **** // void p { if (</pre>	<pre> } , , , , , , , , , , , , , , , , , ,</pre>	B1000001, B1111100, B1111100, B1111100, B011000, B111100, B1001000, B000100, B0011100, B0011100, B1001100, B1001100, B1001100, B1001000, B1111111, B1000001, B0001000, C1111111, B1000001, B1001000, C1111111, B1000001, C1111111, C110100, C1111111, C1111111, C1111111, C1111111, C1111111, C1111111, C1111111, C1111111, C1111111, C111111, C1111111, C1111111, C111111, C111111, C111111, C111111, C111111, C111111, C111111, C111111, C111111, C111111, C111111, C11111, C11111, C11111, C11111, C11111, C1111, C1111, C1111, C1111, C1111, C1111, C1111, C1111, C1111, C1111, C1111, C1111, C1111, C1111, C1111, C1111, C111, C1111, C111,	Billilli, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Boliolio, Bolioloo, Bolio, Bo	B1000000, B111100, B0000100, B100100, B0100100, B0000100, B1000100, B1000100, B1000000, B1000000, B1010000, B1010000, B1001000, B1001100, B000100, B000100, B000100, B000100, B000000, B0000100, B00000, B00000, B000000, B000000, B00000, B00000, B00000, B00000, B00000, B00000, B00000, B0000, B0000, B0000, B000, B000, B000, B000, B000, B000, B000, B000, B000, B	B0000100, B0011000, B1111000, B0111000, B011000, B111100, B0000100, B0100100, B0100000, B111100, B0100000, B0101000, B0101000, B0000000, B0000000, B0000000, B0000100, B0000100, B0000100, B0000100, B0000100, B0000100, B0000100, B0000100, B0000100, B0000100, B0000100, B0000000, B0000100, B0000000, B0000000, B0000100, B000000, B0000000, B0000000, B0000000, B0000000, B0000000, B00000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B000000, B000000, B0000000, B0000000, B000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B000000, B000000, B000000, B000000, B000000, B00000, B000000, B00000, B00000, B00000, B00000, B00000, B00000, B0000, B0000, B0000, B000, B000, B000, B000, B000, B00, B000, B	B1111000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0011100, B0011100, B100100, B000000, B000000, B000000, B000000, B000000, B00000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B0000, B00000, B00000, B0000, B000, B000, B000, B000, B000, B00, B000, B	//////////////////////////////////////	m n o p q r s t u v w x y z { {   } } ~ ~ caratte	976 ************************************
<pre>5, 8 5, 8 4, 8 4, 8 4, 8 4, 8 4, 8 4, 8 5, 8 5, 8 5, 8 4, 8 3, 8 3, 8 3, 8 3, 8 3, 8 3, 8 3, 8 3</pre>		B1000001, B1111100, B1111100, B1111100, B0111000, B1011000, B1011000, B0001000, B0011100, B0011100, B0011100, B1001100, B1001100, B1001100, B1001000, B1111111, B1000001, B1001000, B1111111, S1000001, B1001000, B1111111, S1000001, B1001000, B1111111, S1000001, S1000001, S1000000, S1111111, S1000001, S1000001, S1000000, S1111111, S1000001, S1000001, S1000000, S1111111, S1000001, S1000000, S100000, S100000, S100000, S100000, S100000, S100000, S100000, S100000, S100000, S100000, S100000, S100000, S100000, S100000, S100000, S10000, S10000, S10000, S100000, S1000, S10000, S10000, S10000, S10000, S10000, S10000, S10000, S10000, S1000, S10000, S10000, S10000, S10000, S10000, S10000, S10000, S10000, S1000, S1000, S10000, S10000, S10000, S1	Billilli, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Boliollo, Bolooloo, Boliollo, Bolooloo, Boliollo, Boliol	B1000000, B111100, B0000100, B0100100, B0100100, B0100100, B1000100, B1000100, B1000000, B1000000, B1010000, B1000000, B1000000, B1000000, B000100, B000100, B000100, B000100, B00000, B00000, B000000, B00000, B000000, B000000, B00000, B00000, B000000, B00000, B00000, B00000, B00000, B00000, B00000, B00000, B00000, B00000, B00000, B00000, B00000, B00000, B00000, B00000, B00000, B00000, B0000, B00000, B00000, B000000, B000000, B00000, B00000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B00000, B000000, B000000, B000000, B000000, B000000, B00000, B00000, B00000, B00000, B00000, B0000, B00000, B00000, B00000, B00000, B00000, B00000, B0000, B0000, B0000, B0000, B0000, B000, B000, B000, B000, B000, B000, B000, B000, B000, B000, B000, B000, B00, B000, B0,	B0000100, B0000100, B1111000, B011000, B011000, B111100, B0000100, B0100100, B0100000, B111100, B0100000, B0101000, B0101000, B0000000, B0000000, B0000000, B0000100, B0000100, B0000100, B0000100, B0000100, B0000100, B0000100, B0000100, B0000100, B0000100, B0000100, B0000000, B0000100, B0000000, B0000000, B0000100, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B00000000	B1111000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0011100, B0011100, B0011100, B000000, B000000, B000000, B000000, B00000, B00000, B000000, B00000, B000000, B000000, B00000, B0000, B00000, B00000, B00000, B0000, B0000, B0000, B000, B000, B000, B000, B000, B000, B0,	//////////////////////////////////////	m n 0 p q r s t u v w x y z {   } ~ ~ caratte	Pre ************************************
5, 8 5, 8 4, 8 4, 8 4, 8 4, 8 4, 8 5, 8 5, 8 4, 8 5, 8 5, 8 5, 8 4, 8 5, 8 5, 8 4, 8 3, 8 4, 8 3, 8 4, 8 3, 8 4, 8 4, 8 5, 8 5, 8 5, 8 4, 8 5, 8 5, 8 5, 8 4, 8 5, 8 5, 8 5, 8 5, 8 6 5, 8 6 7, 8 6 7, 8 6 7, 8 7, 8 7, 8 7, 8 7, 8 7, 8 7, 8 7, 8	<pre> } , , , , , , , , , , , , , , , , , ,</pre>	B1000001, B1111100, B1111100, B1111100, B0111000, B1111100, B0011000, B1001000, B0011100, B0011100, B0011100, B1001100, B1001100, B1001100, B1001000, B1111111, B1000001, B1001000, B1111111, B1000001, B1001000, B1111111, B1000001, B1001100, B1111111, B1000001, B1001100, B1111111, B1000001, B1001100, B1111100, B1111100, B1001110, B1001100, B1001100, B1001110, B1000100, B1001110, B1001100, B100110, B100100, B100110, B100100, B100100, B100100, B100100, B100100, B100100, B100110, B100100, B100100, B100110, B100100, B100100, B100100, B100100, B100100, B100100, B100100, B100100, B100100, B100100, B100100, B100100, B100100, B100000, B100000, B100000, B100000, B100000, B100000, B100000, B100000, B100000, B10000, B10000, B10000, B10000, B10000, B10000, B10000, B10000, B10000, B10000, B10000, B10000, B1000, B10000, B10000, B1000, B1000, B1000, B1000, B1000, B1000, B1000, B1000, B1000, B1000, B1000, B1000, B1000, B1000, B1000, B1000, B100, B1000, B1000, B100,	Billilli, Bolooloo, Bolioloo, Bolooloo, Bolio, Bolio	B1000000, B111100, B0000100, B1000100, B0100100, B0000100, B1000100, B1000100, B1000000, B1000000, B1010000, B1010000, B1000001, B0001000, B000100, B0000100, B0000100, B0000100, B0000100, B0000100, B0000000, B0000000, B00000, B000000, B000000, B00000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B00000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, C, C, C, C, C, C, C, C, C, C, C, C, C,	B0000100, B0000100, B111000, B011000, B011000, B0100100, B0100100, B0100000, B1111100, B0100000, B0101000, B0101000, B0000000, B0000000, B0000000, B0000000, B0000100, B0000100, B0000100, B0000100, B0000100, B0000100, B0000100, B0000100, B0000100, B0000100, B0000000, B0000000, B0000100, B000000, B0000000, B00000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B000000, B000000, B000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B000000, B000000, B000000, B000000, B00000, B00000, B00000, B0000, B0000, B0000, B000, B000, B000, B000, B000, B	B1111000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0011100, B0111100, B0111100, B000000, B000000, B000000, B000000, B000000, B00000, B00000, B00000, B00000, B0000, B0000, B0000, B0000, B000, B000, B000, B000, B000, B000, B000, B00,	//////////////////////////////////////	m n o p q r s t u v w x y z {   } ~ caratte ******	9 <i>26</i> ************************************
5, 8 5, 8 4, 8 4, 8 4, 8 4, 8 4, 8 5, 8 5, 8 3, 8 5, 8 5, 8 3, 8 3, 8 4, 8 3, 8 4, 8 3, 8 4, 8 3, 8 4, 8 3, 8 4, 8 4, 8 5, 8 5, 8 8 5, 8 8 7, 8 8 7, 8 8 7, 8 8 7, 8 8 7, 8 8 7, 8 8 7, 8 8 7, 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8 8	</td <td>B1000001, B1111100, B1111100, B1111100, B011000, B1111100, B1001000, B1001000, B001100, B0011100, B1001100, B1001100, B1001100, B1001100, B1001100, B1001000, B1111111, B1000001, B1111111, B1000001, B1111111, B1000001, B1111111, B1000001, B1111111, B1000001, B1111100, B111100, B100000, B1111100, B100000, B1111100, B100000, B1111100, B100000, B1111100, B100000, B1111100, B100000, B100000, B111100, B100000, B100000, B100000, B100000, B100000, B100000, B10000, B100000, B100000, B100000, B100000, B100000, B100000, B100000, B100000, B10000, B10000, B100000, B10000, B10000, B10000, B10000, B10000, B10000, B10000, B10000, B10000, B10000, B10000, B10000, B100000, B10000, B10000, B10000, B10000, B10000, B10000, B10000, B10000, B10000, B10000, B10000, B10000, B100000, B10000, B10000, B10000, B10000, B10000, B100000, B100000, B100000, B100000, B100000, B10000, B10000, B100000, B100000, B100000, B100000, B10000, B10000, B10000, B10000, B10000, B100000, B10000, B10000, B10000, B10000, B10000, B1000, B1000, B1000, B1000, B1000, B1000, B1000, B1000, B1000, B1000, B1000, B1000, B1000, B1000, B1000, B100, B1000, B1000, B100, B1000, B1000, B100, B1000, B1000, B100, B1000, B100, B</td> <td>Billilli, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bololoo, Bololoo, Bol</td> <td>B1000000, B111100, B0000100, B1000100, B0100100, B0100100, B1000100, B1000100, B1000000, B1000000, B1010000, B1001000, B1001000, B0000000, B000100, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B00000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B00000, B00000, B00000, B00000, B0000, B0000, B0000, B000, B000, B000, B000, B000, B000, B000, B000, B000, B000, B000, B000, B000, B000, B000, B000, B000, B000, B00, B000, B0,</td> <td>B0000100, B0000100, B1111000, B011000, B011000, B0100100, B0100100, B0100000, B1111100, B0100000, B0101000, B0101000, B0000000, B0000000, B0000000, B0000000, B0000100, B0000100, B0000100, B0000100, B0000100, B0000100, B0000100, B0000100, B0000100, B0000100, B0000000, B0000100, B0000000, B0000000, B0000100, B0000000, B0000000, B0000000, B0000000, B0000100, B000000, B0000000, B0000000, B000000, B000000, B0000000, B000000, B000000, B000000, B00000, B000000, B000000, B00000, B00000, B00000, B00000, B00000, B00000, B0000, B000, B0000, B0000, B000, B000, B000, B000, B000, B000, B000, B</td> <td>B1111000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0111100, B0111100, B000000, B000000, B000000, B000000, B00000, B00000, B00000, B00000, B00000, B0000, B0000, B0000, B000, B000, B000, B000, B000, B000, B0,</td> <td>//////////////////////////////////////</td> <td>m n o p q r s t u v w x y z {   } ~ caratte ******</td> <td>97e ************************************</td>	B1000001, B1111100, B1111100, B1111100, B011000, B1111100, B1001000, B1001000, B001100, B0011100, B1001100, B1001100, B1001100, B1001100, B1001100, B1001000, B1111111, B1000001, B1111111, B1000001, B1111111, B1000001, B1111111, B1000001, B1111111, B1000001, B1111100, B111100, B100000, B1111100, B100000, B1111100, B100000, B1111100, B100000, B1111100, B100000, B1111100, B100000, B100000, B111100, B100000, B100000, B100000, B100000, B100000, B100000, B10000, B100000, B100000, B100000, B100000, B100000, B100000, B100000, B100000, B10000, B10000, B100000, B10000, B10000, B10000, B10000, B10000, B10000, B10000, B10000, B10000, B10000, B10000, B10000, B100000, B10000, B10000, B10000, B10000, B10000, B10000, B10000, B10000, B10000, B10000, B10000, B10000, B100000, B10000, B10000, B10000, B10000, B10000, B100000, B100000, B100000, B100000, B100000, B10000, B10000, B100000, B100000, B100000, B100000, B10000, B10000, B10000, B10000, B10000, B100000, B10000, B10000, B10000, B10000, B10000, B1000, B1000, B1000, B1000, B1000, B1000, B1000, B1000, B1000, B1000, B1000, B1000, B1000, B1000, B1000, B100, B1000, B1000, B100, B1000, B1000, B100, B1000, B1000, B100, B1000, B100, B	Billilli, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bolooloo, Bololoo, Bololoo, Bol	B1000000, B111100, B0000100, B1000100, B0100100, B0100100, B1000100, B1000100, B1000000, B1000000, B1010000, B1001000, B1001000, B0000000, B000100, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B00000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B00000, B00000, B00000, B00000, B0000, B0000, B0000, B000, B000, B000, B000, B000, B000, B000, B000, B000, B000, B000, B000, B000, B000, B000, B000, B000, B000, B00, B000, B0,	B0000100, B0000100, B1111000, B011000, B011000, B0100100, B0100100, B0100000, B1111100, B0100000, B0101000, B0101000, B0000000, B0000000, B0000000, B0000000, B0000100, B0000100, B0000100, B0000100, B0000100, B0000100, B0000100, B0000100, B0000100, B0000100, B0000000, B0000100, B0000000, B0000000, B0000100, B0000000, B0000000, B0000000, B0000000, B0000100, B000000, B0000000, B0000000, B000000, B000000, B0000000, B000000, B000000, B000000, B00000, B000000, B000000, B00000, B00000, B00000, B00000, B00000, B00000, B0000, B000, B0000, B0000, B000, B000, B000, B000, B000, B000, B000, B	B1111000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0111100, B0111100, B000000, B000000, B000000, B000000, B00000, B00000, B00000, B00000, B00000, B0000, B0000, B0000, B000, B000, B000, B000, B000, B000, B0,	//////////////////////////////////////	m n o p q r s t u v w x y z {   } ~ caratte ******	97e ************************************
<pre>5, 8 5, 8 4, 8 4, 8 4, 8 4, 8 4, 8 4, 8 5, 8 5, 8 5, 8 3, 8 3, 8 1, 8 3, 8 4, 8 }; // **** // void p {     if (         c -==         memco         mem</pre>	<pre></pre>	B1000001, B1111100, B1111100, B1111100, B0111000, B1111100, B1001000, B1001000, B0001000, B0011100, B1001100, B1001100, B1001100, B1001100, B1001100, B1001000, B1111111, B1000001, B1001000, B1111111, Colump(max)	Billilli, Bolooloo, Bolioloo, Boliol	B1000000, B111100, B0000100, B1000100, B0100100, B0000100, B1000100, B1000000, B1000000, B1010000, B1010000, B1001000, B1001000, B000100, B000100, B00000, B00000, B000000, B000000, B00000, B000000, B000000, B00000, B00000, B00000, B00000, B00000, B00000, B0000, B000, B0000, B000, B000, B000, B000, B000, B000, B000, B000, B000, B000, B00, B000, B0	B0000100, B0000100, B1111000, B0111000, B011000, B0100100, B0100100, B0100000, B1111100, B0100000, B0101000, B0101000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000100, B0000100, B0000100, B0000100, B0000100, B0000100, B0000100, B0000000, B0000100, B0000000, B0000100, B0000000, B0000000, B0000000, B0000100, B000000, B0000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B00000, B00000, B00000, B0000, B0000, B0000, B000, B000, B000, B000, B000, B000, B000, B000, B00, B000, B00, B00, B00, B00, B00, B00, B00, B00, B00, B00, B00, B00, B00, B0, B	BI111000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0000000, B0011100, B0111100, B1001000, B000000, B000000, B000000, B000000, B000000, B00000, B000000, B000000, B000000, B000000, B000000, B000000, B00000, B0000, B0000, B00000, B000, B000, B000, B000, B000, B00, B000, B0,	//////////////////////////////////////	m n o p q r s t u v w x y z {   } ~ caratte *	ere ***********************************

```
for (int i = 0; i <= Buf7219[0]; i++)
 {
   delay(shift speed);
   m.shiftLeft(false, false);
 }
}
11
   void printStringWithShift(char* s, int shift speed)
{
                                //inizializza il contatore di caratteri
 contatore = 0;
 while (*s != 0)
 {
   printCharWithShift(*s, shift speed);
   s++;
   contatore++;
                                   // incrementa di 1 il contatore di caratteri della frase
   if (contatore > contacaratteri)
                                   // se e' stata visualizzata tutta la frase
    return;
                                   // esce dal loop di esposizione
 }
}
11
11
void setup()
{
 m.init(); // inizializza le matrici
 m.setIntensity(3); // definisce l'intensita' luminosa
 // Serial.begin(9600); // inizializza il collegamento con il monitor seriale sul pc
 BT.begin(9600); // inizializza la comunicazione seriale emulata dalla connessione Bluetooth
}
11
11
() gool biov
{
  // if (Serial.available()) //se e' in arrivo qualcosa dalla tastiera del pc
                           // se e' arrivato qualcosa dalla porta seriale
 if (BT.available())
 {
   for (contacaratteri = 0; contacaratteri <= lunghezzafrase; contacaratteri++) //innesca il loop
// di ricezione dei caratteri in arrivo - receiving loop for incoming characters
   {
     // string [contacaratteri] = Serial.read(); //legge (inserisce in "string" cio' che e' stato
                                             battuto sulla tastiera
//se la frase e' terminata (se nulla e' in arrivo)
     // if (!(Serial.available ()))
     string [contacaratteri] = BT.read();
                                            //legge (inserisce in "string" cio' che e' stato
// battuto sulla tastiera bluetooth - insert in "string" what is arrived from bluetooth keyboard
  if (!(BT.available ())) //se la frase e' terminata (se nulla e' in arrivo) - if nothing is
incoming
        return;
                                             // esce dal loop di ricezione
   }
 }
 else
 {
   delay(100);
   m.shiftLeft(false, true);
   printStringWithShift (string, 150); //lancia la routine di visualizzazione della frase pervenuta
 }
}
```