

Pustili smo rob karakterja, širine enega kvadrata, ker ga imajo tudi ostale črke (razen malih črk z repom, ki sega do dna).

2. Odločite se, kateri od simbolov bo za definirano grafiko predstavljal π . (Recimo, da tisti, ki se nanaša na prko P tako, da se bo po pritisku na tipko P v grafičnem modu pojavil π).
3. Pripravite v spominu nov vzorec. Vsak takšen grafični prikaz - vzorec, ki se shrani kot 8 števil, eno za vsako vrsto. Vsako od teh osem števil lahko napišete kot BIN, za katerim je osem 0 ali 1; 0 za papir, 1 za črnilo. Tako je sedaj osem števil za naš karakter:

BIN 00000000

BIN 00000000

BIN 00000010

BIN 00111100

BIN 01010100

BIN 00010100

BIN 00000000

(Če poznate binarna števila, boste opazili, da se BIN uporablja za pisanje števil v binarnem sistemu namesto v decimalnem sistemu).

Teh osem števil se shrani v spominu na osmih mestih in vsako ima svoj naslov. Naslov prvega bajta ali skupine osmih števil, je USR "P" (P zaradi tega, ker je to tisto, kar smo izbrali v 2), naslov drugega je USR "P"+1, in vse do osmega, ki ima naslov USR "P"+7. USR označuje tukaj funkcijo, ki pretvarja argument stringa v naslov prvega bajta v spominu, za odgovarjajočo samostalno definirano grafiko. String-argument mora biti en karakter, ki je lahko uporabljen za samodefinirajočo grafiko ali ustrezno črko (veliko ali malo).

Obstoja še ena možnost uporabe USR, kadar je njen argument število, vendar si jo bomo ogledali pozneje.

Čeprav tega ne razumete, bo naslednji program to napravil za vas: