

```

1 4:STRINGS.PRG v0.2.0a, 7 августа 2022
2 Знакогенератор, литеры, строки и работа с ними
3
4 Авторское право (с) Васильев Илья Владимирович, г. Москва
5 Это свободная программа, защищённая авторским правом
6 Распространяется под лицензией GNU GPL v3 или старше
7
8 OPTION STRICT 'Все переменные должны быть объявлены
9 OPTION DEFINT 'Массивы по умолчанию целые
10
11 DIM MNEMO$(256) 'Мнемоника ПМК
12 COPY MNEMO$,@Mnemonic 'Считать мнемонику
13
14 DIM PrgStr$(65535) 'Память строк
15 CONST #STREMPY=&hf8ffffff 'Пустая строка
16
17 - Разобраться с литерой "-" (30fc)
18
19 Символы ПМК
20
21 МК-161 SB4 Юникод
22
23 18 e21e 2194 ↑ UPWARDS ARROW
24 19 e21f 2195 ↗ RIGHTWARDS ARROW
25 20 e220 2200 DOUBLE PRIME
26 21 e221 2201 PRIME
27 22 e222 2202 MODIFIER LETTER SMALL X
28 23 e223 2203 MODIFIER LETTER SMALL Y
29 24 e224 2204 DIVISION SIGN
30 25 e225 2205 GREATER-THAN OR SLANTED EQUAL TO
31 26 e226 2206 NOT EQUAL TO
32 27 e227 2207 MULTIPLICATION SIGN
33 28 e228 2208 CLOCKWISE GAPPED CIRCLE ARROW
34 29 e229 2209 LEFT RIGHT ARROW
35 30 e22a 2210 GREEK SMALL LETTER PI
36 31 e22b 2211 SUPERSCRIPT TWO
37 32 e22c 2212 DOWNWARDS ARROW WITH CORNER LEFTWARDS
38 33 e22d 2213 DEGREE SIGN
39 34 e22e 2214 SQUARE ROOT
40 35 e22f 2215 LOGICAL OR
41 36 e230 2216 CIRCLED PLUS
42
43 Убрано:
44 e2e6 2227 Δ LOGICAL AND
45
46 Альтернативные коды
47 B1 2265 GREATER-THAN OR EQUAL TO
48
49 Преобразовать код Юникода (МК-261) в литеру SB4
50 - можно добавить "." вместо отсутствующих литер
51 COMMON DEF GetUnicode$(C)
52 CASE C
53 WHEN "↑" THEN RETURN "e21e 2194"
54 WHEN "↗" THEN RETURN "e21f 2195"
55 WHEN "〃" THEN RETURN "e220 2200"
56 WHEN "′" THEN RETURN "e221 2201"
57 WHEN "×" THEN RETURN "e222 2202"
58 WHEN "×" THEN RETURN "e223 2203"
59 WHEN "÷" THEN RETURN "e224 2204"
60 WHEN "≧" THEN RETURN "e225 2205"
61 WHEN "≠" THEN RETURN "e226 2206"
62 WHEN "×" THEN RETURN "e227 2207"
63 WHEN "Δ" THEN RETURN "e2e6 2227"
64 WHEN "<" THEN RETURN "e22f 2215"
65 WHEN ">" THEN RETURN "B1 2265"
66 OTHERWISE RETURN CHR$(C)
67 ENDCASE
68
69 Преобразовать строку Юникода (МК-261) в SB4
70 COMMON DEF FromUnicode$(A$)
71 VAR I,B$
72 FOR I=0 TO LAST(A$)
73 PUSH B$,GetUnicode$(ASC(A$(I)))
74 NEXT I
75 RETURN B$
76
77 Подготовить и закодить строку для вывода на индикатор
78 - можно измерять и ограничивать длину строки в пикселях
79 COMMON DEF StrQ$(A$)
80 IF LEN(A$)<10 THEN
81 RETURN CHR$(34)+FromUnicode$(A$)+CHR$(34)
82 ELSE
83 RETURN CHR$(34)+FromUnicode$(LEFT$(A$,8))+CHR$(&h2026)
84 ENDIF
85
86 Преобразовать латинские буквы в строчные
87 COMMON DEF ToLower$(A$)
88 VAR I,B$,C$,D=ASC("a")-ASC("A")
89 FOR I=0 TO LAST(A$)
90 C$=A$(I)
91 IF C$>="A" && C$<="Z" THEN PUSH B$,CHR$(ASC(C$)+D) ELSE PUSH B$,C$
92 NEXT I
93 RETURN B$
94
95 Перевести код операции в мнемонику
96 COMMON DEF Mnemonic$(iCop)
97 RETURN MNEMO$(iCop)
98
99 Считать и декодировать шаг из памяти программ
100 COMMON DEF StepDecode N OUT H$,T$,flgBrkP
101 H$=""
102 T$=""
103 flgBrkP=#FALSE
104 IF N>=0 && N<&h10000 THEN
105 VAR S=VAR("1:Prm") [N],S0=S>>>24
106 IF S0<&hf0 THEN 'число
107 H$="":T$=ToLower$(HEX$(S>>>16,4)+": "+HEX$(S AND 65535,4))
108 ELSE
109 IF (S0 AND 8)=0 THEN flgBrkP=#TRUE:INC S0,8
110
111 IF S0=&hf8 THEN 'Строка ++сделать интеллектуальной
112 H$="":T$=StrQ$(StrGet$(S))
113 ELSEIF S0=&h255 THEN 'Команда
114 S0=(S>>>16) AND 255:S=S AND 65535
115 IF S0=&h255 THEN S0=S AND 255:S=65535 'КОП в младшем байте
116 H$=ToLower$(HEX$(S0,2)):T$=Mnemonic$(S0)
117 CASE S0

```

## STRINGS.PRG v0.2a (2/2)

```

000125 H$=ToLower$(HEX$(S0,2)):T$=Mnemonic$(S0)
000126 CASE S0
000127 WHEN &hf4:: WHEN &hf51:: WHEN &hf53:: WHEN &hf57:: WHEN &hf58:: WHEN &hf59::
000128 WHEN &hf5a:: WHEN &hf5b:: WHEN &hf5c:: WHEN &hf5d:: WHEN &hf5e:: WHEN &hf5f::
000129 WHEN &hf6:: WHEN &hf6b:: WHEN &hf6c::
000130 PUSH T$, ToLower$(HEX$(S,4))
000131 OTHERWISE
000132 IF S!=65535 THEN PUSH T$,"→"+ToLower$(HEX$(S,4))
000133 ENDCASE
000134 ENDDIF 'Команда
000135
000136 ENDDIF 'S0 таг
000137 ENDDIF 'N в 0000..ffff
000138 ENDD
000139
000140
000141 'Очистить память строк
000142 ' - вызывается из PrgClear
000143 COMMON DEF StrClear
000144 FILL PrgStr$,
000145 ENDD
000146
000147
000148 'Записать строку в память строк и вернуть её код
000149 COMMON DEF StrLink(A$)
000150 IF A$!= THEN
000151 VAR N=Find(PrgStr$, "")
000152 IF N>=0 THEN PrgStr$(N)=A$:RETURN &hf8ff0000+N
000153 VAR("0:ERROR")=12 'Переполнение памяти строк
000154 ENDDIF
000155 RETURN #STREMPY
000156 ENDD
000157
000158
000159 'Убрать строку из памяти строк
000160 COMMON DEF StrUnlink N
000161 IF (N AND &hf7ff0000)==&hf0ff0000 THEN
000162 VAR N0=N AND 65535
000163 IF N0<65535 THEN PrgStr$(N0)=" "
000164 ENDDIF
000165 ENDD
000166
000167
000168 'Считать строку по её коду
000169 COMMON DEF StrGet$(N)
000170 IF (N AND &hf7ff0000)==&hf0ff0000 THEN
000171 VAR N0=N AND 65535
000172 IF N0<65535 THEN RETURN PrgStr$(N0)
000173 ENDDIF
000174 RETURN " "
000175 ENDD
000176
000177
000178 @Mnemonic
000179 DATA "0"
000180 DATA "1"
000181 DATA "2"
000182 DATA "3"
000183 DATA "4"
000184 DATA "5"
000185 DATA "6"
000186 DATA "7"
000187 DATA "8"
000188 DATA "9"
000189 DATA "A"
000190 DATA "B"
000191 DATA "C"
000192 DATA "D"
000193 DATA "E"
000194 DATA "F"
000195 DATA "0"
000196 DATA "1"
000197 DATA "2"
000198 DATA "3"
000199 DATA "4"
000200 DATA "5"
000201 DATA "6"
000202 DATA "7"
000203 DATA "8"
000204 DATA "9"
000205 DATA "A"
000206 DATA "B"
000207 DATA "C"
000208 DATA "D"
000209 DATA "E"
000210 DATA "F"
000211 DATA "0"
000212 DATA "1"
000213 DATA "2"
000214 DATA "3"
000215 DATA "4"
000216 DATA "5"
000217 DATA "6"
000218 DATA "7"
000219 DATA "8"
000220 DATA "9"
000221 DATA "A"
000222 DATA "B"
000223 DATA "C"
000224 DATA "D"
000225 DATA "E"
000226 DATA "F"
000227 DATA "0"

```