MUAZ TA$TEMEL

This assembly program find the numbers with color guidence. The codes of the program explained below.

Some messages for explain the rules of the program and guidence.

```
INCLUDE Irvine32.inc

.data

;msg1 byte "*****basarili ici nex stage gecis****",0
;msg2 byte "esit degil new_input ici",0
msg2 byte "Welcome to Color Number Game. You are in Stage1! You Can 6 Guess in 2 minute!",
msg3 byte "Congratulations You are in Stage2! You Can 8 Guess in 2 minute !",0
msg4 byte "GAME OVER!!! Much Trial!!!",0
msg5 byte "Congratulations You are in Stage3! You Can 10 Guess in 2 minute !",0
msg6 byte "!!!Congratulations You Win!!!",0
msg7 byte "Number of Try: ",0
msg8 byte "renklendirme cikis ",0
msg9 byte "Game Over!!! Time Left!!! ",0

```

The initialization of the variables and registers.

```


random5 DWORD 5 DUP (?)
guess5 DWORD 5 DUP (0)
guesscolor5 DWORD 5 DUP (?)

random6 DWORD 6 DUP (?)
guess6 DWORD 6 DUP (?)
guesscolor6 DWORD 6 DUP (?)

random7 DWORD 7 DUP (?)
guess7 DWORD 7 DUP (?)
guesscolor7 DWORD 7 DUP (?)

; k==0  b==1 y==2


stage5count DWORD ?
stage6count DWORD ?
stage7count DWORD ?

stage5 DWORD 5
stage6 DWORD 6
stage7 DWORD 7

```

This program has 3 main loop for three stages, and sub loop for;

- Creating ramdom numbers lengths of 5,6,7 and taking in a register

- Taking inputs from user
- Another loop for comparing inputs and ramdom characters. It take numbers the number 0 for red color, number 1 for White color, number 2 for green color.
- Another loop for changing the guess arrays numbers due to 0,1 and 2's.

The General stage5 loop

```
122
123             add stage5count,1   ; input sayılıyor
124             cmp stage5count,7
125             je game_over
126
127
128             call crlf
129             mov edx, offset msg7
130             call writestring
131
132             mov eax, DWORD ptr stage5count ; deneme sayacı
133             call writedec
134             call crlf
135             cmp stage5count,7
136             je game_over
137
138             push ecx ; sure control
139             call readch5
140             call getmseconds
141             sub eax,timing5
142             mov timing25,eax
143
144             cmp timing25,120000
145             ja timeleft
146             pop ecx
147
148         call crlf
149         call table5_
150
151
152             call crlf
153             ;call readch5
```

Erasing Guess color for new input.

```
186
187  new_input5:
188                  ;call crlf
189                  ;mov edx, offset msg2
190                  ;call writestring
191
192
193                  call crlf
194                  ;call readch
195                  ;call crlf
196                  jmp guess_color_bosalt5
197
198      guess_color_bosalt5:
199
200                  mov esi,0
201                  push ecx
202                  mov ecx,stage5
203
204      gk5:
205                  mov eax,0
206                  mov DWORD PTR guesscolor5[esi*4],eax
207                  inc esi
208                  loop gk5
209                  pop ecx
210
211      ;call compare
212      ;call sonucyaz
213
214      dec cx
215      jne stage_5
216
```

Teable procedur for dispalying the user inputs in a table.

```
493  ;****************
494  table_ proc
495
496  mov esi,0
497  mov ecx, 66
498
499  tb:
500          mov eax,table5 [esi*4]
501          call writechar
502          inc esi
503          loop tb
504
505
506  ret
507  table_ endp
508
509  ; alınan sayıyı arraye atma
510
511
```

İntroanime Procedure for displaying introduction animation.

```
570  ;*******************************begin input
571  introanime PROC
572  mov edx,0
573
574                  mov esi,0
575                  call clrscr
576                  mov ecx,13
577                  loop111:
578                  mov introdelay,ecx
579                  mov ecx,48
580                  loop211:
581                  mov eax,q1[esi*4]
582                  call writeChar
583                  mov eax,15
584                  call delay
585                  inc esi
586                  loop loop211
587                  mov ecx,introdelay
588                  call crlf
589                  loop loop111
590                  mov eax,6000
591                  call delay
592
593  ret
594  introanime ENDP
595  ;*************************
```

Procedure for reading char from user.

```
29
30  ;**********************readchar proc
31  readch5 proc
32
33  mov esi ,0
34  mov ecx ,stage5
35  mov edi,24
36
37  l15:
38      mov eax,0
39      top5:
40      call readchar
41      cmp eax,48
42      jb ring5
43      cmp eax,57
44      ja ring5
45      jmp correct5
46      ring5:
47      mov eax,7
48      call writechar
49      jmp top5
50      correct5:
51      mov DWORD PTR randomtable5[edi*4],eax
52      add edi,4
53      sub eax,48
54      mov DWORD PTR guess5[esi*4],eax
55      call writedec
56      inc esi
57      ;add edi,2
58      loop l15
59
```

Creating random numbers between 0 and 10.

```
765
766  ret
767  randomproc5 endp
768  ;************************
769
770
771  ; random sayı alma
772  ;************************ random number proc
773  randomproc6 proc
774
775  call randomize
776  mov ecx,stage6
777  mov esi,0
778  mov edi,28
779  L16:                        ;random sayısı oluşur
780      mov eax,10
781      call randomrange
782      mov DWORD PTR random6[esi*4],eax
783      ;call writedec
784      add eax,48
785      mov DWORD PTR randomtable6[edi*4],eax
786      add edi,4
787      inc esi
788
789  loop L16
790
791  ret
792  randomproc6 endp
793  ;************************
794
```

First part of compare procedure for color of the numbers

```
;******************** compare procedure
compare5 PROC

pushad

        mov eax,0
        mov ebx,0
        mov esi,0
        mov edi,0
        mov ecx,stage5
guess_array5:

        mov esi,0
        push ecx
        mov ecx, stage5
random_array5:


        mov ebx,0
        mov eax,0

        mov ebx,DWORD PTR random5[esi*4]  ; eax random sayının array  adresi
        mov eax,DWORD PTR guess5[edi*4]   ; ebx girilen sayının array adresi


        cmp eax,ebx
        je  yesilmibeyazmi5
        jmp red 5
```

First Part of the compare and write procedure.

```
1115  ;****************** renklendirme
1116  sonucyaz5 PROC
1117
1118  mov eax,0
1119  mov ebx,0
1120  mov esi,0
1121  mov edi,24
1122  mov ecx,stage5
1123
1124  renkbelirle:
1125
1126
1127              cmp DWORD PTR guesscolor5[esi*4],1
1128              je white_
1129              ja green_
1130              jb red_
1131
1132  white_:
1133          mov eax,white + (black * 16)
1134          call SetTextColor
1135          mov eax,DWORD PTR guess5[esi*4]
1136          call writedec
1137          add eax,48
1138          ;mov DWORD PTR table5[edi*4],eax
1139
1140          mov eax,yellow + (black * 16)
1141          call SetTextColor
1142          add edi,4
1143          inc esi
1144          cmp esi,5
1145          je ext_
1146          dec ecx
```

This procedure writes the number which compared and colored numbers in the table on window.

```
1367
1368  ;****************
1369  tablerandom5_ proc
1370
1371  call randomproc5
1372
1373
1374  mov esi,0
1375  mov ecx, 66
1376
1377  tb5:
1378          mov eax,randomtable5 [esi*4]
1379          call writechar
1380          inc esi
1381          loop tb5
1382
1383  ret
1384  tablerandom5_ endp
1385
1386  ;********************************** alınan sayıyı arraye atma
1387
```