



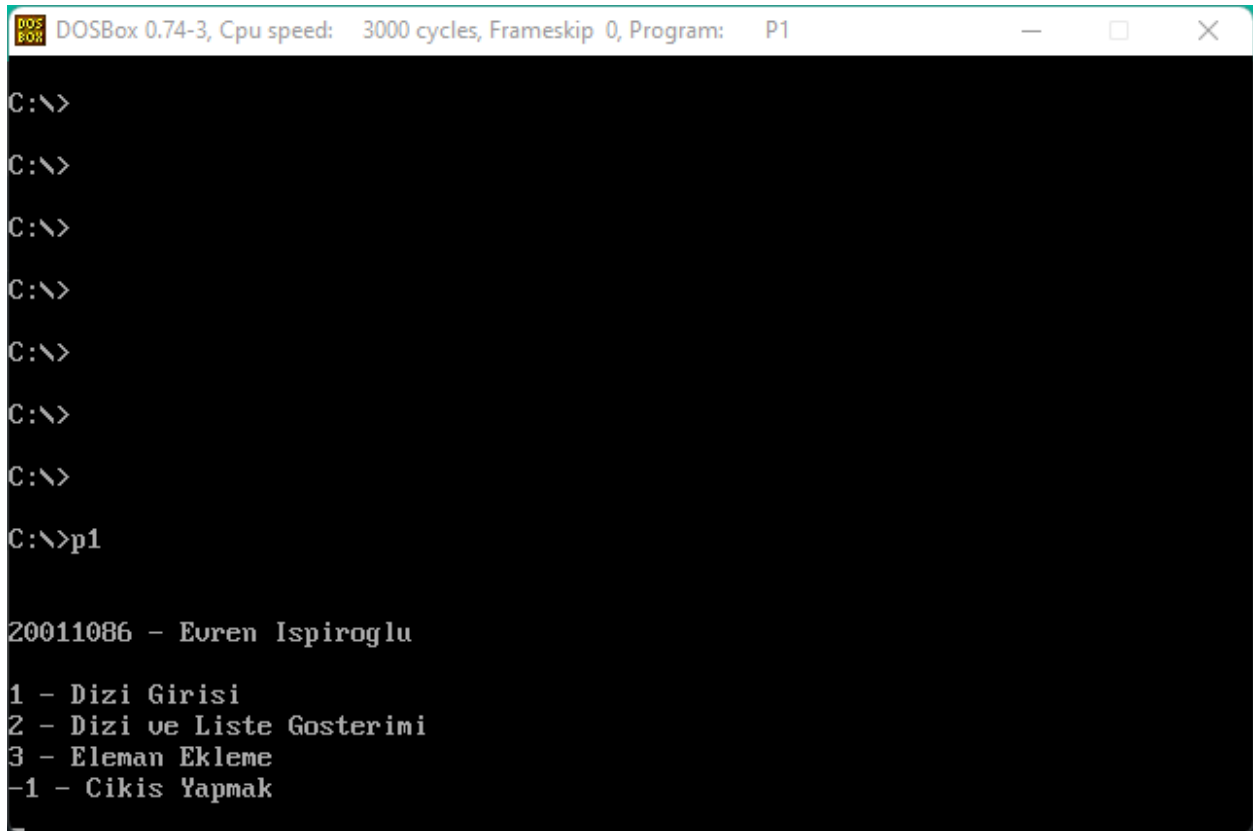
Evren İspiroğlu

20011086

GR : 2

Öğr.Gör. Furkan Çakmak

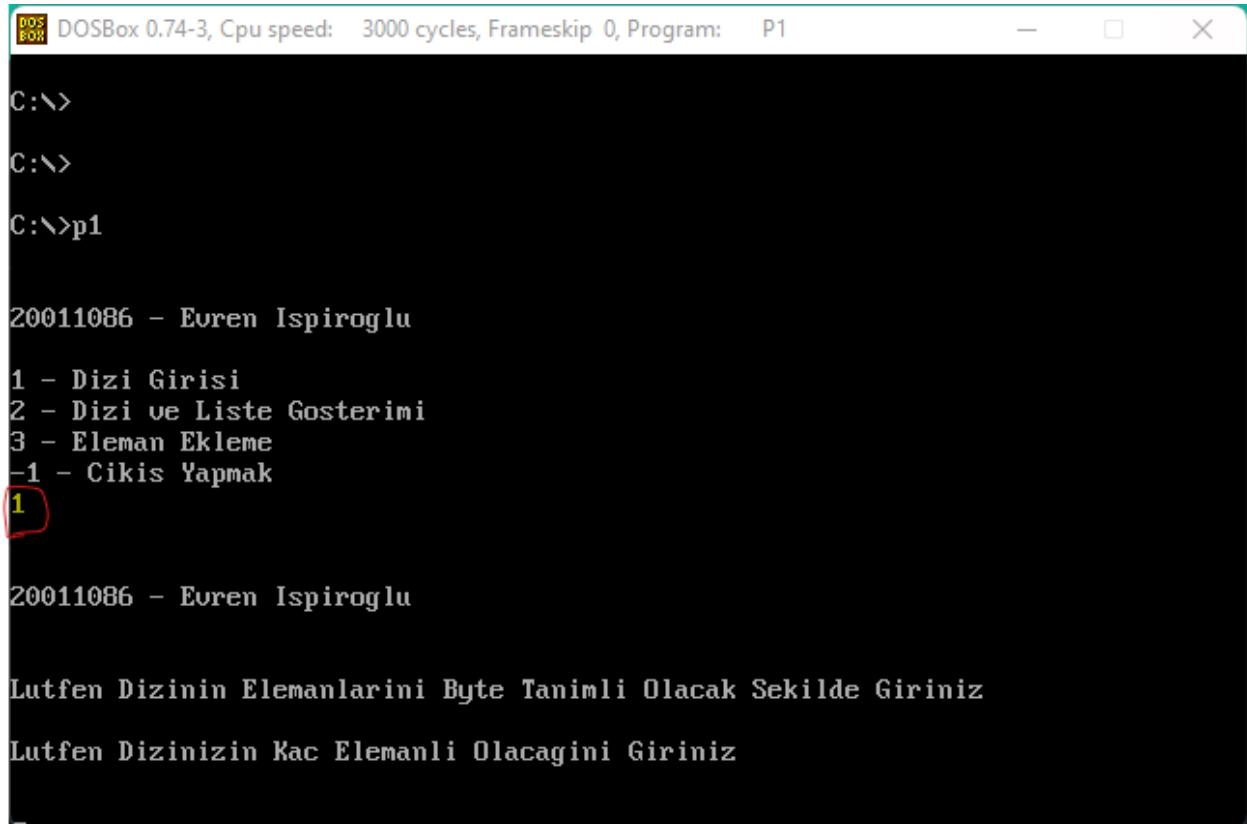
Programın Çalıştırılması



DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: P1

```
C:\>  
C:\>  
C:\>  
C:\>  
C:\>  
C:\>  
C:\>  
C:\>  
C:\>p1  
  
20011086 - Evren Ispiroglu  
  
1 - Dizi Girisi  
2 - Dizi ve Liste Gosterimi  
3 - Eleman Ekleme  
-1 - Cikis Yapmak  
_
```

Menüden 1. Alt Menüye Gidilmesi



```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: P1

C:\>

C:\>

C:\>p1

20011086 - Evren Ispiroglu

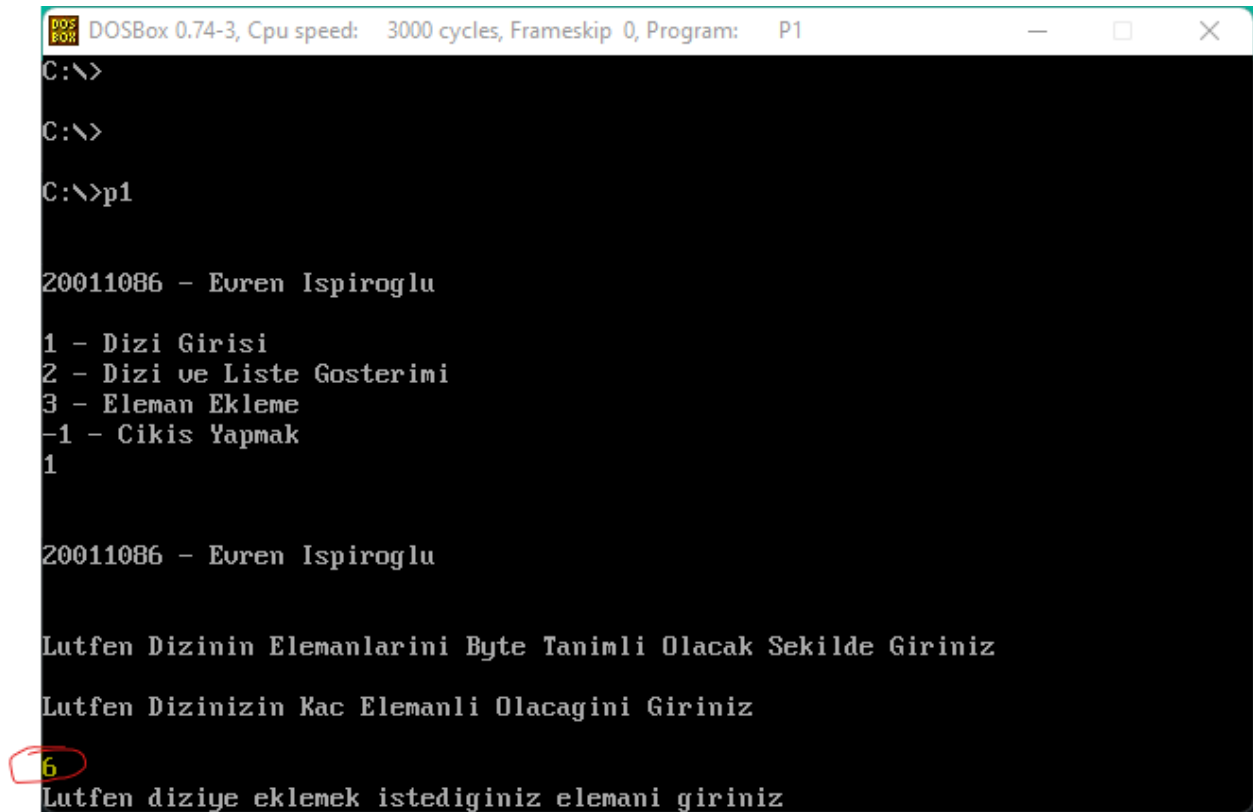
1 - Dizi Girisi
2 - Dizi ve Liste Gosterimi
3 - Eleman Ekleme
-1 - Cikis Yapmak
1

20011086 - Evren Ispiroglu

Lutfen Dizinin Elemanlarini Byte Tanimli Olacak Sekilde Giriniz

Lutfen Dizinizin Kac Elemanli Olacagini Giriniz
```

Dizinin Eleman sayısının alınması



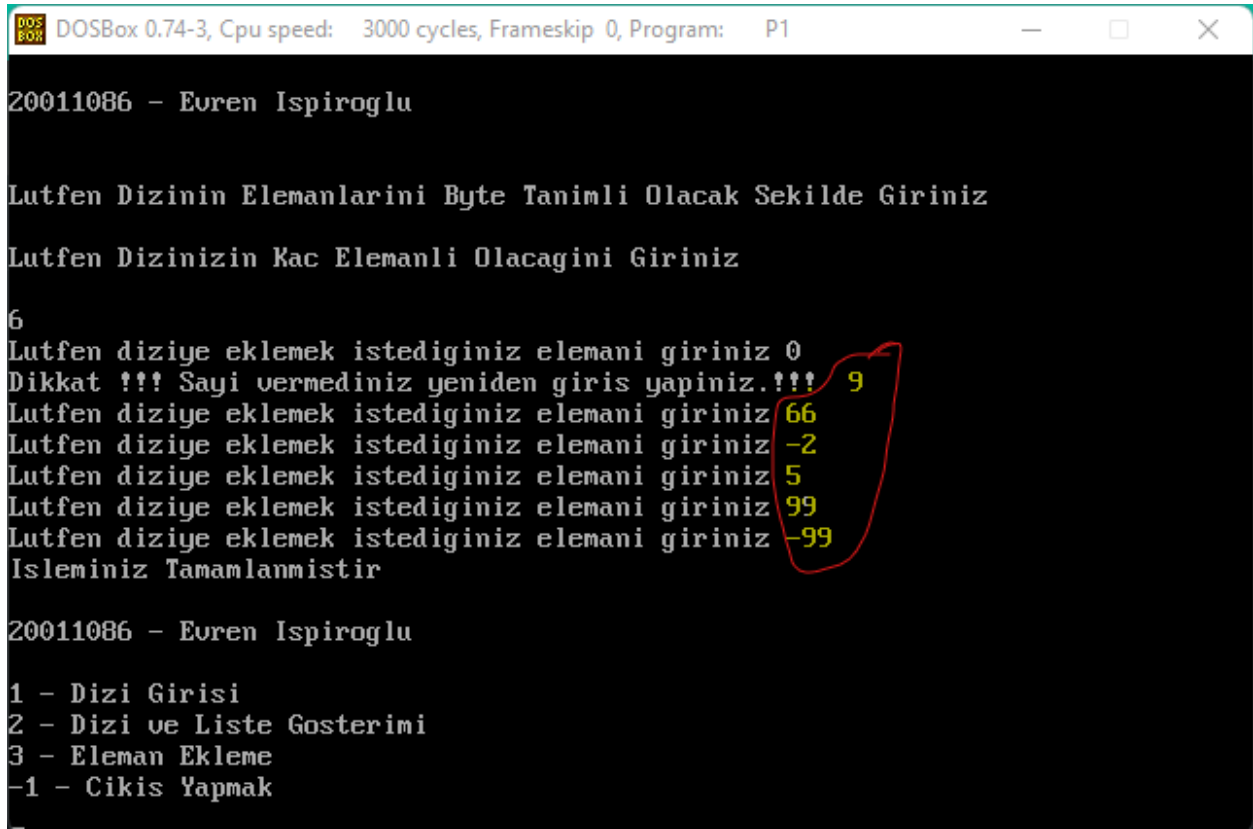
```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: P1
C:\>
C:\>
C:\>p1

20011086 - Euren Ispiroglu
1 - Dizi Girisi
2 - Dizi ve Liste Gosterimi
3 - Eleman Ekleme
-1 - Cikis Yapmak
1

20011086 - Euren Ispiroglu

Lutfen Dizinin Elemanlarini Byte Tanimli Olacak Sekilde Giriniz
Lutfen Dizinizin Kac Elemanli Olacagini Giriniz
6
Lutfen diziye eklemek istediginiz elemani giriniz
```

Dizinin elemanlarının tek tek girilmesi



```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: P1

20011086 - Euren Ispiroglu

Lutfen Dizinin Elemanlarini Byte Tanimli Olacak Sekilde Giriniz

Lutfen Dizinizin Kac Elemanli Olacagini Giriniz

6
Lutfen diziye eklemek istediginiz elemani giriniz 0
Dikkat !!! Sayi vermediniz yeniden giris yapiniz.!!! 9
Lutfen diziye eklemek istediginiz elemani giriniz 66
Lutfen diziye eklemek istediginiz elemani giriniz -2
Lutfen diziye eklemek istediginiz elemani giriniz 5
Lutfen diziye eklemek istediginiz elemani giriniz 99
Lutfen diziye eklemek istediginiz elemani giriniz -99
Isleminiz Tamamlanmistir

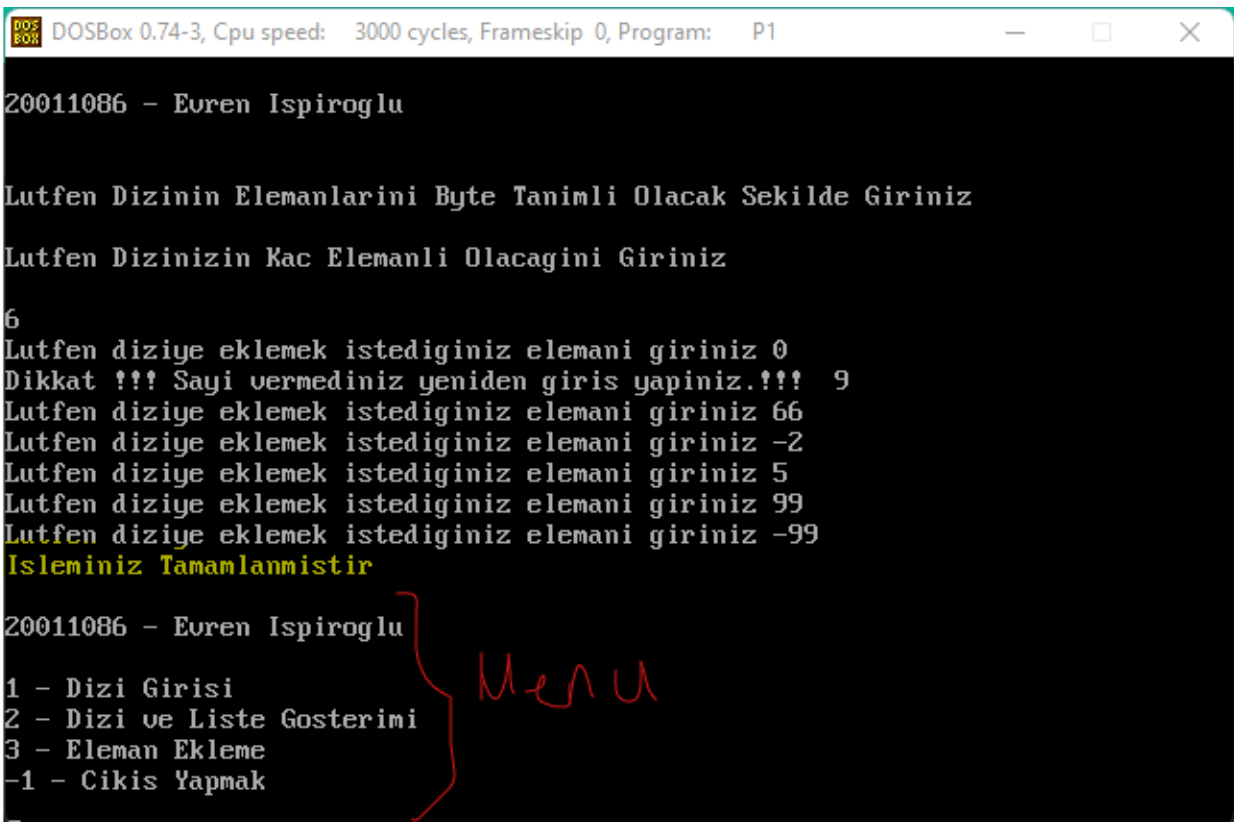
20011086 - Euren Ispiroglu

1 - Dizi Girisi
2 - Dizi ve Liste Gosterimi
3 - Eleman Ekleme
-1 - Cikis Yapmak
```

*Aradaki hata mesajı ekran görüntüsü almaya çalışırken girilen
yanlış inputu silmemden dolayı oluşmuştur.*

*Hata mesajı kitapta bize sağlanan GET fonksiyonlarından
gelmektedir.*

Dizinin girilmesinin tamamlanması ve yeni menü seçimi için input beklenmesi

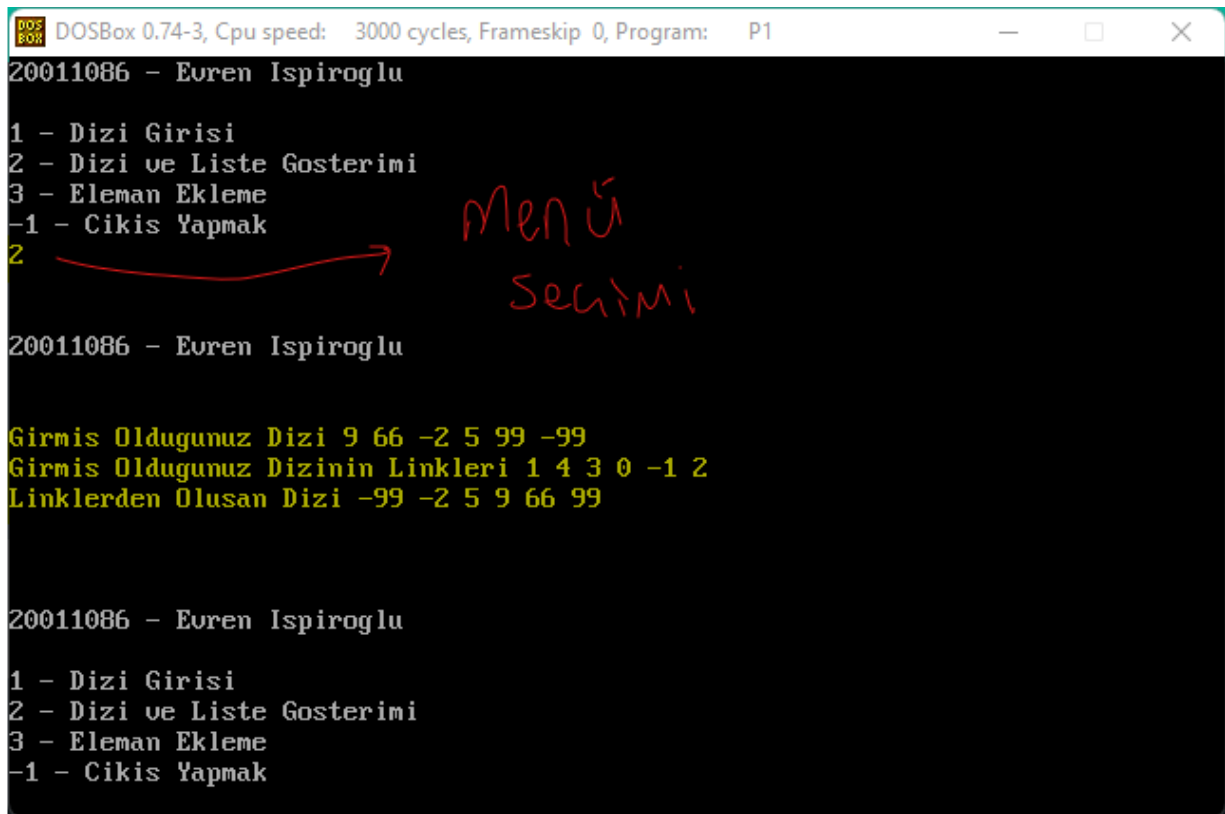


```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: P1
20011086 - Euren Ispiroglu

Lutfen Dizinin Elemanlarini Byte Tanimli Olacak Sekilde Giriniz
Lutfen Dizinizin Kac Elemanli Olacagini Giriniz
6
Lutfen diziye eklemek istediginiz elemani giriniz 0
Dikkat !!! Sayi vermediniz yeniden giris yapiniz.!!! 9
Lutfen diziye eklemek istediginiz elemani giriniz 66
Lutfen diziye eklemek istediginiz elemani giriniz -2
Lutfen diziye eklemek istediginiz elemani giriniz 5
Lutfen diziye eklemek istediginiz elemani giriniz 99
Lutfen diziye eklemek istediginiz elemani giriniz -99
Isleminiz Tamamlanmistir
20011086 - Euren Ispiroglu
1 - Dizi Girisi
2 - Dizi ve Liste Gosterimi
3 - Eleman Ekleme
-1 - Cikis Yapmak
```

Menu

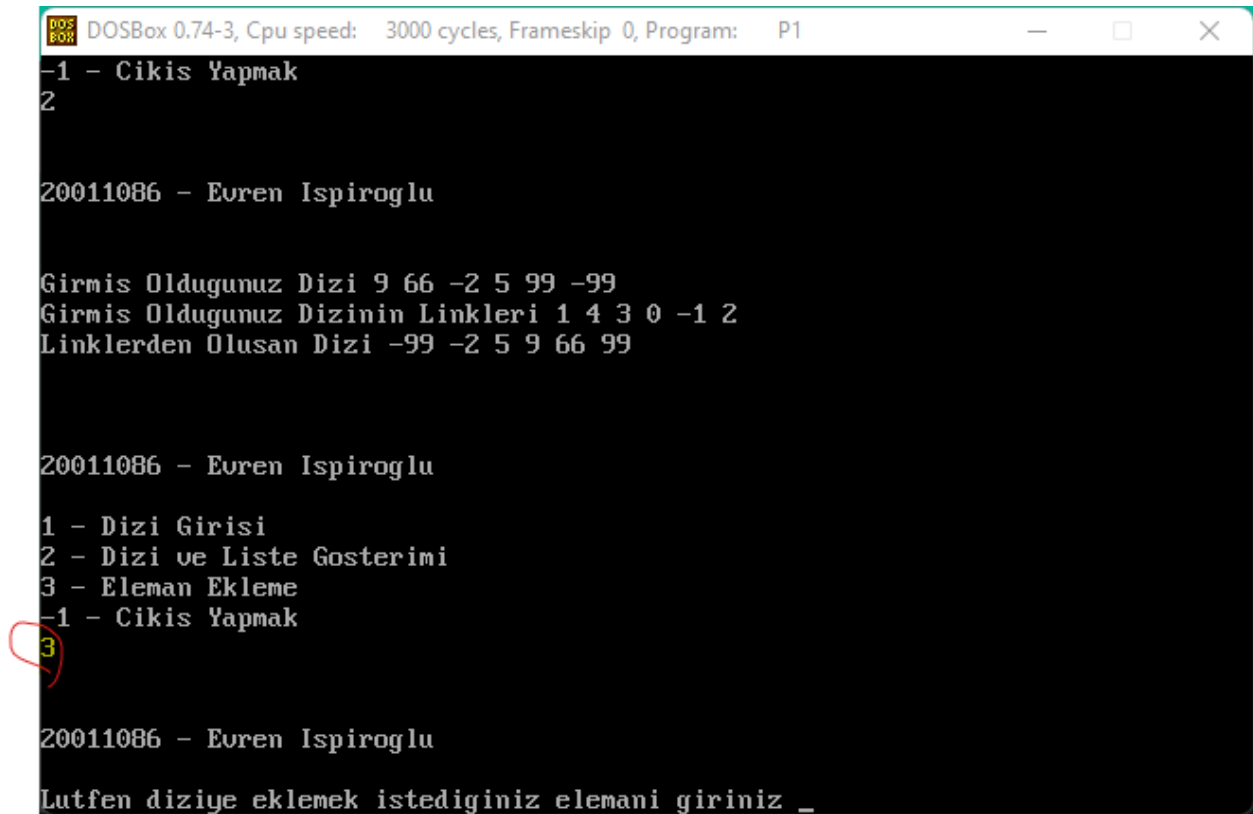
Menüde 2. Alt menüye geçilip dizi elemanlarının ekrana basılması



```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: P1
20011086 - Euren Ispiroglu
1 - Dizi Girisi
2 - Dizi ve Liste Gosterini
3 - Eleman Ekleme
-1 - Cikis Yapmak
2
Menü Seçimi
20011086 - Euren Ispiroglu
Girmis Oldugunuz Dizi 9 66 -2 5 99 -99
Girmis Oldugunuz Dizinin Linkleri 1 4 3 0 -1 2
Linklerden Olusan Dizi -99 -2 5 9 66 99
20011086 - Euren Ispiroglu
1 - Dizi Girisi
2 - Dizi ve Liste Gosterini
3 - Eleman Ekleme
-1 - Cikis Yapmak
```

Ekrana basım tamamlandıktan sonra
tekrar input bekleniyor.

Menude 3. Alt menünün seçilmesi



```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: P1
-1 - Cikis Yapmak
2

20011086 - Euren Ispiroglu

Girmis Oldugunuz Dizi 9 66 -2 5 99 -99
Girmis Oldugunuz Dizinin Linkleri 1 4 3 0 -1 2
Linklerden Olusan Dizi -99 -2 5 9 66 99

20011086 - Euren Ispiroglu

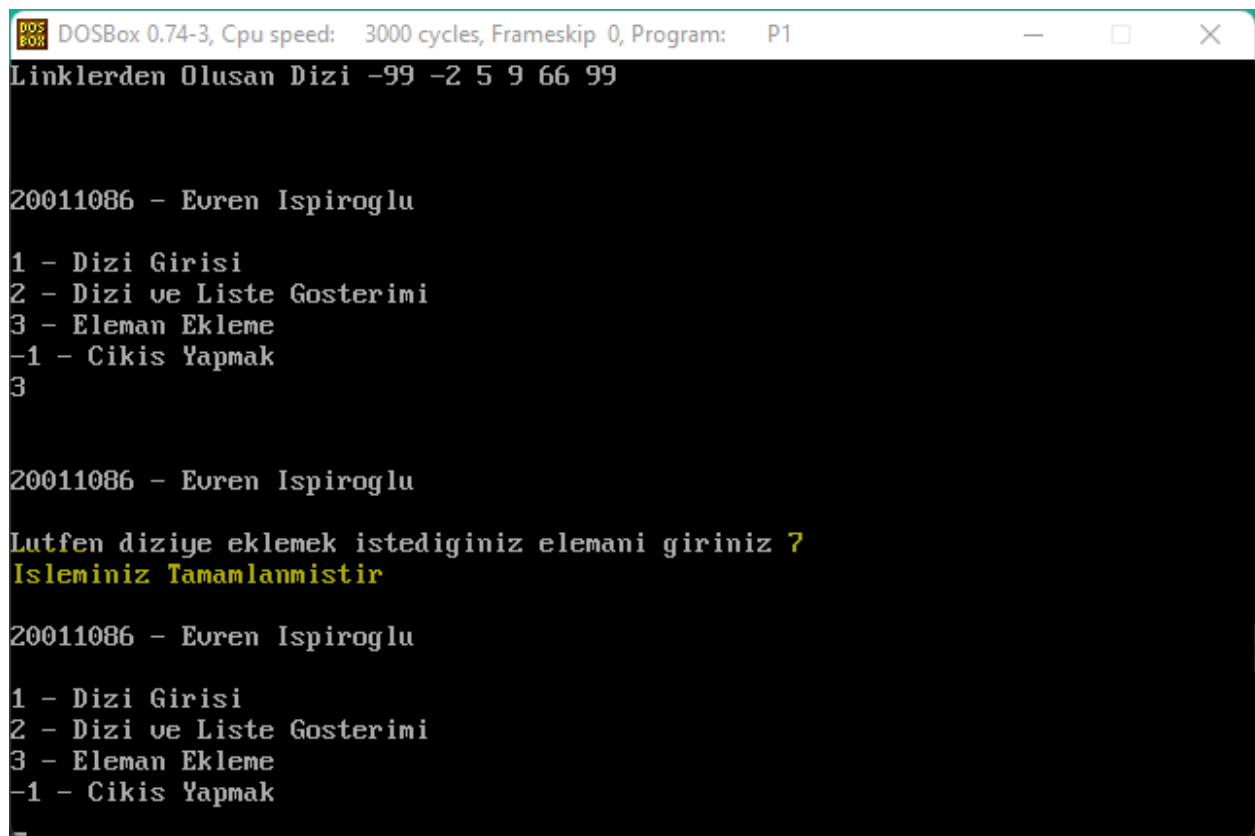
1 - Dizi Girisi
2 - Dizi ve Liste Gosterimi
3 - Eleman Ekleme
-1 - Cikis Yapmak
3

20011086 - Euren Ispiroglu

Lutfen diziye eklemek istediginiz elemani giriniz _
```

Diziye eklenecek elemanın
inputunun beklenmesi

Inputun alınıp insert işleminin yapılması



```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: P1
Linklerden Olusan Dizi -99 -2 5 9 66 99

20011086 - Euren Ispiroglu

1 - Dizi Girisi
2 - Dizi ve Liste Gosterimi
3 - Eleman Ekleme
-1 - Cikis Yapmak
3

20011086 - Euren Ispiroglu

Lutfen diziye eklemek istediginiz elemani giriniz ?
Isleminiz Tamamlanmistir

20011086 - Euren Ispiroglu

1 - Dizi Girisi
2 - Dizi ve Liste Gosterimi
3 - Eleman Ekleme
-1 - Cikis Yapmak
```

Menüde 2. Alt Menünün Seçilip dizi çıktılarının alınması

```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: P1
20011086 - Euren Ispiroglu

1 - Dizi Girisi
2 - Dizi ve Liste Gosterimi
3 - Eleman Ekleme
-1 - Cikis Yapmak
2

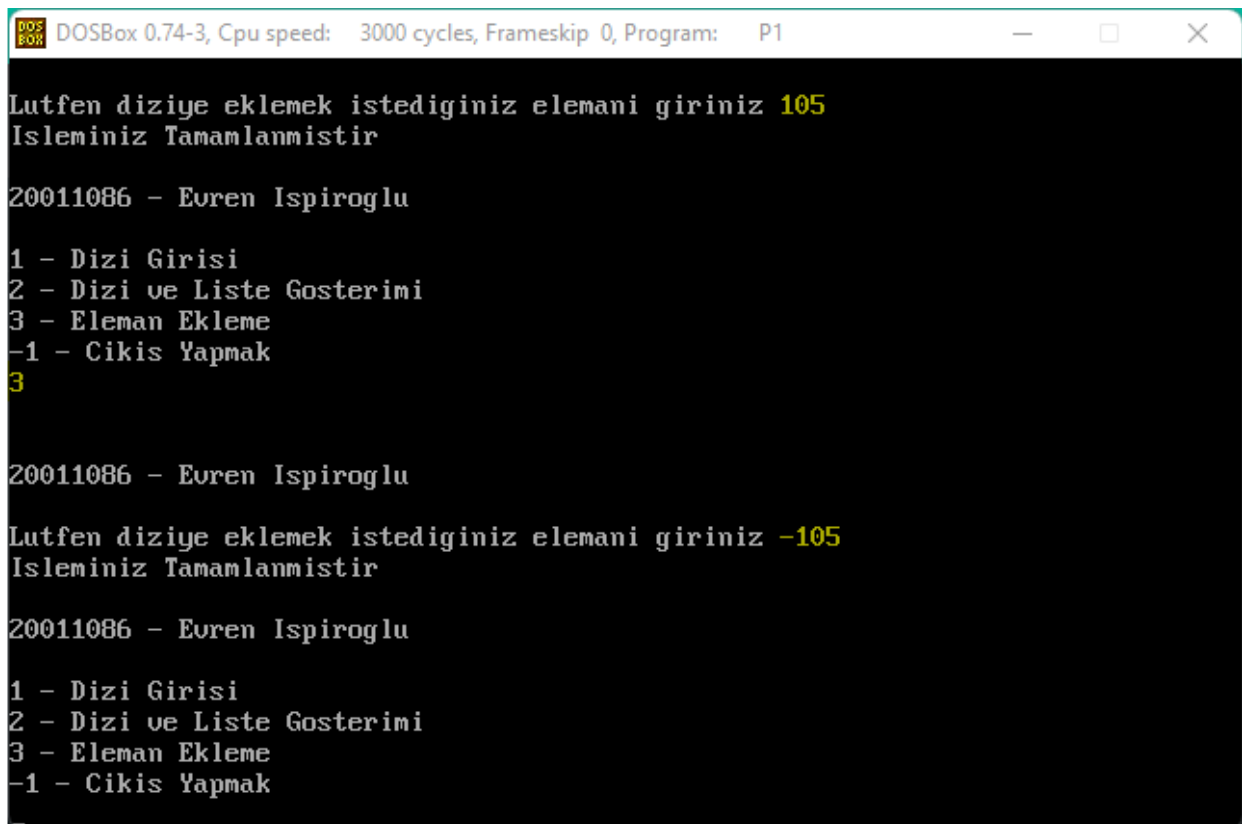
20011086 - Euren Ispiroglu

Girmis Oldugunuz Dizi 9 66 -2 5 99 -99 7
Girmis Oldugunuz Dizinin Linkleri 1 4 3 6 -1 2 0
Linklerden Olusan Dizi -99 -2 5 7 9 66 99

20011086 - Euren Ispiroglu

1 - Dizi Girisi
2 - Dizi ve Liste Gosterimi
3 - Eleman Ekleme
-1 - Cikis Yapmak
```

Ekleme işlemlerinin max ve min uç değerleri için de yapılması



```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: P1

Lutfen diziye eklemek istediginiz elemani giriniz 105
Isleminiz Tamamlanmistir

20011086 - Euren Ispiroglu

1 - Dizi Girisi
2 - Dizi ve Liste Gosterimi
3 - Eleman Ekleme
-1 - Cikis Yapmak
3

20011086 - Euren Ispiroglu

Lutfen diziye eklemek istediginiz elemani giriniz -105
Isleminiz Tamamlanmistir

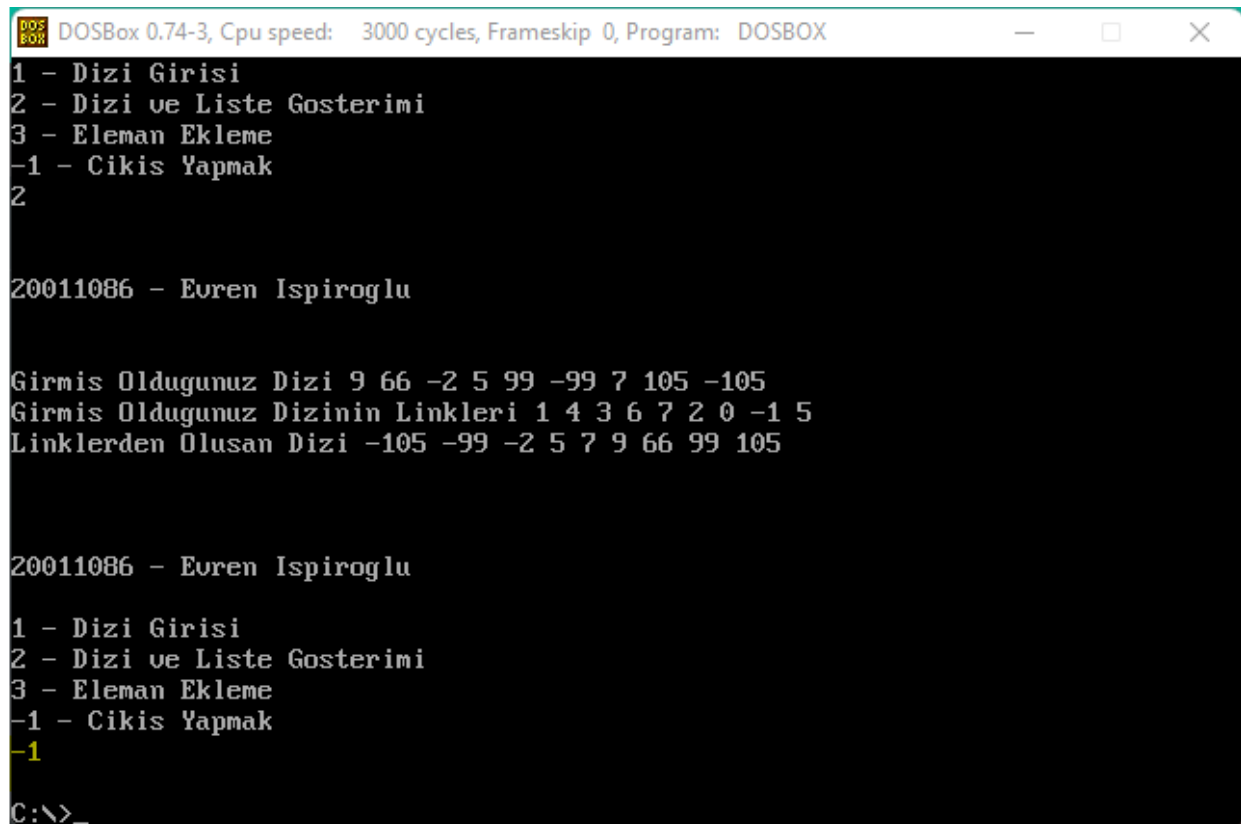
20011086 - Euren Ispiroglu

1 - Dizi Girisi
2 - Dizi ve Liste Gosterimi
3 - Eleman Ekleme
-1 - Cikis Yapmak
_
```

Bu islemlerden sonra dizi çıktılarının alınması

```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: P1
20011086 - Euren Ispiroglu
1 - Dizi Girisi
2 - Dizi ve Liste Gosterimi
3 - Eleman Ekleme
-1 - Cikis Yapmak
2
20011086 - Euren Ispiroglu
Girmis Oldugunuz Dizi 9 66 -2 5 99 -99 7 105 -105
Girmis Oldugunuz Dizinin Linkleri 1 4 3 6 7 2 0 -1 5
Linklerden Olusan Dizi -105 -99 -2 5 7 9 66 99 105
20011086 - Euren Ispiroglu
1 - Dizi Girisi
2 - Dizi ve Liste Gosterimi
3 - Eleman Ekleme
-1 - Cikis Yapmak
```

-1 Inputu girilip Programdan Çıkılması



```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DOSBOX
1 - Dizi Girisi
2 - Dizi ve Liste Gosterimi
3 - Eleman Ekleme
-1 - Cikis Yapmak
2

20011086 - Euren Ispiroglu

Girmis Oldugunuz Dizi 9 66 -2 5 99 -99 7 105 -105
Girmis Oldugunuz Dizinin Linkleri 1 4 3 6 7 2 0 -1 5
Linklerden Olusan Dizi -105 -99 -2 5 7 9 66 99 105

20011086 - Euren Ispiroglu

1 - Dizi Girisi
2 - Dizi ve Liste Gosterimi
3 - Eleman Ekleme
-1 - Cikis Yapmak
-1
C:\>_
```

Kodun Açıklanması

1. Alt Menü

```
1  CMP AL, 1          ;; 1 kontrolu. 1 girilirse dizi initialize ediliyor.
2                      JNE not_1
3
4                      CALL printInfo      ;; Alt menude ogrenci bilgilerimin bastirilmesi.
5
6                      CALL getArray       ;; Ek bir yordam yardimiyla dizinin kullanicidan alinmasi
7                      LEA AX, Done
8                      CALL PUT_STR        ;; Kullanciya islemin tamamlandiginin bildirilmesi
9
```

Menü başında öğrenci bilgilerimin *printInfo* yordamı ile bastırılması

Ve *getArray* yardamı sayesinde dizinin alınması.

Sonrasında da işlemin bittiğine dair kullanıcının bilgilendirilmesi.

2. Alt Menü

```
1  CMP AL, 2          ;; 2 kontrolu. 2 girilirse didi, linkler dizisi ve linklerin takibinden oluş
   an
2      JNE not_2        ;; dizinin yazdirilmesi.
3      CALL printInfo   ;; Alt menude ogrenci bilgilerimin bastirilmesi.
4
5
6
7      MOV AX, LF
8      CALL PUTC
9
10     LEA AX, MSG1
11     CALL PUT_STR      ;; Kullanicinin girdigi dizinin basitirilmesi
12     CALL printArray
13
14     LEA AX, MSG2
15     CALL PUT_STR      ;; Olusturulan link dizisinin basitirilmesi.
16     CALL printList
17
18     LEA AX, MSG3
19     CALL PUT_STR      ;; Link dizisinin takibiyle olusan dizinin bastirilmesi.
20     CALL listToArray
21
22     MOV AX, LF
23     CALL PUTC
```

printInfo yordamı yardımı ile öğrenci bilgilerimin bastırılması

printArray, *printList*, *listToArray* yordamları yardımı ile elimizdeki listelerin ekrana bastırılması. Sonrasında da DOSBOX üstünde güzel görünmesi için alt satıra inilmesi.

3. Alt Menü

```
1  CMP AL, 3          ;; 3 kontrolu. 3 Girilirse diziye yeni eleman eklenecek
2
3                      JNE start
4
5                      CALL printInfo      ;; Alt menude ogrenci bilgilerimin yazdirilmasi.
6
7                      CALL insertInput    ;; Ek yordam yardimiyla kullanicidan alinan inputun insert e
   dilmesi.
8
9                      LEA AX, Done        ;; Islemin bittigine dair kullanicinin bilgilendirilmesi.
10                     CALL PUT_STR
```

printInfo yordamı yardımı ile öğrenci bilgilerimin bastırılması

insertInput yordamı yardımı ile kullanıcıdan input alınıp alınan değere göre işlemlerin yapılması.

İşlemin başarı ile sonuclandırığının çıktısı.

getArray Yordamı

```
1  getArray PROC NEAR                ;; Kullanıcıdan diziye alacak yordam.
2
3      PUSH AX
4      PUSH CX
5      PUSH SI
6
7      MOV AX, LF                    ;; Kullanıcıyı yönlendirmek için
8      CALL PUTC                     ;; Gerekli mesajların bastırılması
9      LEA AX, MSG4
10     CALL PUT_STR
11     MOV AX, LF
12     CALL PUTC
13     MOV AX, LF
14     CALL PUTC
15     LEA AX, MSG5
16     CALL PUT_STR
17     MOV AX, LF
18     CALL PUTC
19     MOV AX, LF
20     CALL PUTC
21
22     CALL GETN                     ;; Kullanıcıdan dizideki eleman
23     MOV n, AX                     ;; Sayısının alınması
24
25     MOV CX, AX
26     XOR SI, SI
```

Yordam içinde gerekli yönlendirmeleri yapmak için ekrana çıktıların yapılması.

Kullanıcıdan N değerinin alındıktan sonra CX'e atanıp loopun başlaması. (Yordamın devamı aşağıda devam etmektedir.)

```
1  getLoop:
2      LEA AX, MSG6                ;; Dizin elemanlarının tek tek alınması
3      CALL PUT_STR
4      XOR AX, AX
5      CALL GETN
6      MOV mainArray[SI], AX
7      CALL insertNode             ;; Ek yordam yardımı ile her eleman alındığında
8                                  ;; linkli listeye eleman eklenmesi
9
10     ADD SI, 2
11     LOOP getLoop
12
13     POP SI
14     POP CX
15     POP AX
16     RET
17
18 getArray ENDP
```

Alınan her eleman dizide bir sonraki indexte eklenirken InsertNode yordamı yarımı ile linkliListede insert işleminin yapılması.

InsertNode Yordamı

```
1  insertNode proc NEAR                ;; Linkli liste için insert yapan yordam
2      PUSH SI
3      PUSH DI
4      PUSH BX
5      PUSH AX
6
7      CMP AX, min                      ;; Girilen elemanın linkli listede
8      JGE NOTLESS                     ;; En bastaki elemandan küçük mü kontrolü
9      MOV min, AX
10     MOV BX, startIndex               ;; Küçük olması durumunda linkli listede
11     mov indexList[SI], BX           ;; Head node'unun değişimi
12     MOV startIndex, SI
13     JMP link_end
14 NOTLESS:
```

Eklenen elemanın node'un en başındaki elemandan küçük mü kontrolü. Eğer küçükse yeni min o değere atanıyor ve head node'u o değer indexi ile değiştiriliyor.

```
1  MOV DI, startIndex
2
3
4      cmp AX, MAX                      ;; Girilen elemanın linkli listede
5      JLE NOTGREATER                  ;; En sondaki elemandan büyük mü kontrolü
6      MOV MAX, AX
7      MOV AX, -1
8
9      ip                               ;; Büyük olması durumunda listenin sonuna kadar ilerley
10     wLoop:                           ;; Son elemanın değiştirilmesi.
11         CMP AX, indexList[DI]
12         JE true1
13         MOV DI, indexList[DI]
14         JMP wLoop
15 true1:
16     MOV indexList[DI], SI
17     MOV indexList[SI], -1
18     JMP link_end
```

Eklenen elemanın listenin en büyük elemanından büyük mü kontrolü. Eğer büyükse elemanın index değerinin listenin en sonuna eklenmesi.

```
1  NOTGREATER:
2      CMP indexList[DI], -1      ;; Girilen elemanın en büyük ya da en küçük olmaması durumunda
3      JE link_end              ;; linkedList'te indexin bir yeri gösterdiğinden emin olduktan
4      sonra
5          MOV BX, indexList[DI]
6          CMP AX, mainArray[BX]  ;; Girilen elemanın kendisinden küçük en büyük
7          JLE true2              ;; Elemana kadar ulaşılması ve
8          MOV DI, indexList[DI]  ;; sonraki node'ların Düzenlenmesi
9          JMP NOTGREATER
10
11 true2:
12     MOV AX, indexList[DI]
13     MOV indexList[SI], AX
14     MOV indexList[DI], SI
15
16 link_end:
17     CMP SI, 0
18     JNE ENDD
19     MOV AX, mainArray[SI]
20     MOV MAX, AX
21     MOV MIN, AX
22
23 endd:
24     POP AX
25     POP BX
26     POP DI
27     POP SI
28     RET
29 insertNode ENDP
```

Eklenen elemanın arada bir değer olması durumunda kendisinden küçük en büyük elemana kadar gidilip oradaki index değerinin güncellenmesi.

Link_end label'ında yapılan control ilk defa eleman eklenmesi durumunda min ve max değerlerini o elemana göre güncelleyip linkli listeyi kullanıma hazır hale getiriyor.

printArray Yordamı

```
1  printArray PROC NEAR    ;; Kullanici dizisinin bastirilmesi icin kullanılan yordam.
2      PUSH SI
3      PUSH CX
4      PUSH AX
5
6      XOR SI, SI
7      MOV CX, n
8      XOR AX, AX
9
10     printLoop:
11
12         MOV AX, mainArray[SI]
13         CALL PUTN
14         MOV AX, ' '      ;; Elemanlar arasında BOSLUK karakterinin koyulması
15         CALL PUTC
16         ADD SI, 2
17         LOOP printLoop
18
19     MOV AX, LF
20     CALL PUTC
21     POP AX
22     POP CX
23     POP SI
24     RET
25 printArray ENDP
```

Dizi üzerinde gezilerek her elemanın PUTN yordamı yardımı ile ekrana basılması ve her eleman arasına bir bosluk karakteri basılması.

PrintList Yordamı

```
1  printList PROC NEAR      ;; Linkli listenin bastirilmesi icin kullanan yordam.
2      PUSH BX
3      PUSH SI
4      PUSH CX
5      PUSH AX
6
7      XOR SI, SI
8      MOV CX, n
9      XOR AX, AX
10
11     printLoop2:
12
13         MOV AX, indexList[SI]
14         CMP AX, -1
15         JE minusOne
16         SHR AX, 1          ;; SHR yapilmasinin sebebi kullanicidan alinan dizinin WORD tanimli olm
17                             ;; linkli listede tutulan indexlerin kullanicida kafa karistirmamasi ac
18                             ;; isindan
19         minusOne:          ;; SADECE print isleminde bir manipulasyon yapilip
20                             ;; Kullanicida ust seviye dillerdeki gibi bir index bilgisinin verilmes
21                             ;; i.
22         CALL PUTN
23         MOV AX, ' '
24         CALL PUTC
25         ADD SI, 2
26         LOOP printLoop2
27
28     MOV AX, LF
29     CALL PUTC
30     POP AX
31     POP CX
32     POP SI
33     POP BX
34     RET
35 printList ENDP
```

indexList dizisinde bizim asıl kullandığımız array'in index değerleri tutulmakta. Array'imiz WORD tanımlı olduğundan Tutulan index değerleri yüksek dillerdeki gibi 1 – 2 – 3 şeklinde değil 2 – 4 – 6 şeklinde ilerlemektedir. Bu kafa karışıklığını atlatmak adına indexler -1 olmadığı sürece index değerlerini ikiye bölüp ekrana öyle yazdırıyoruz.

listToArray Yordamı

```
1  listToArray PROC NEAR    ;; Linkli listede tutulan index degerlerinden
2      PUSH SI              ;; Yola cikilarak kullanicidan alinan dizideki degerlerin
3      PUSH CX              ;; Sirali bir bicimde yazdirilmesini saglayan yordam.
4      PUSH AX
5      PUSH BX
6
7      XOR SI, SI
8      MOV CX, n
9      XOR AX, AX
10     MOV BX, startIndex
11
12     printLoop3:
13         MOV AX, mainArray[BX]
14         CALL PUTN
15         MOV AX, ' '
16         CALL PUTC
17         MOV BX, indexList[BX]
18         ADD SI, 2
19         LOOP printLoop3
20
21     MOV AX, LF
22     CALL PUTC
23
24     POP BX
25     POP AX
26     POP CX
27     POP SI
28     RET
29 listToArray ENDP
```

indexListte tutulan index deęlerini kullanıp ana listedeki deęerlere erişip ekrana sıralı bir liste yazdırmamızı saęlayan bir algoritma.

InsertInput Yordamı

```
1  insertInput PROC NEAR    ;; Kullanıcıdan alınan alinan bir inputun diziye eklenmesi
2      PUSH AX              ;; Ve linkli listenin buna gore tekrar duznelenmesini
3      PUSH CX              ;; Saglayan yordam.
4      PUSH BX
5      PUSH SI
6
7
8      MOV AX, n
9      INC n
10     MOV CX, AX            ;; Diziye eklenecek olan elemanın
11     SAL AX, 1             ;; Indisinin ele edilmesi.
12     MOV SI, AX            ;; Word tabanlı bir dizi oldugundan
13     LEA AX,MSG6           ;; Eleman sayisi * 2 bizim eklenecek indisimiz olackatir.
14     CALL PUT_STR
15     CALL GETN
16     MOV mainArray[SI], AX ;; Dizinin sonuna elemanın eklenmesi.
17
18     CALL insertNode       ;; Onceden yazilmis instert yordamini tekrar
19                           ;; Kullanarak linkliListenin duzenlenmesi.
20
21     POP SI
22     POP BX
23     POP CX
24     POP AX
25
26     RET
insertInput ENDP
```

Dizi tanımlandıktan sonra kullanıcıdan alınan inputun linkliListede insert edilmesi ve dizinin en sonuna eklenmesini sağlayan yordam.