



ALT SEVİYE PROGRAMLAMA ÖDEV RAPORU 2

ÖĞRENCİ AD: Ertuğrul ŞENTÜRK

ÖĞRENCİ NO: 18011028

ÖĞRENCİ MAIL: mdesenturk@gmail.com

l1118028@std.yildiz.edu.tr

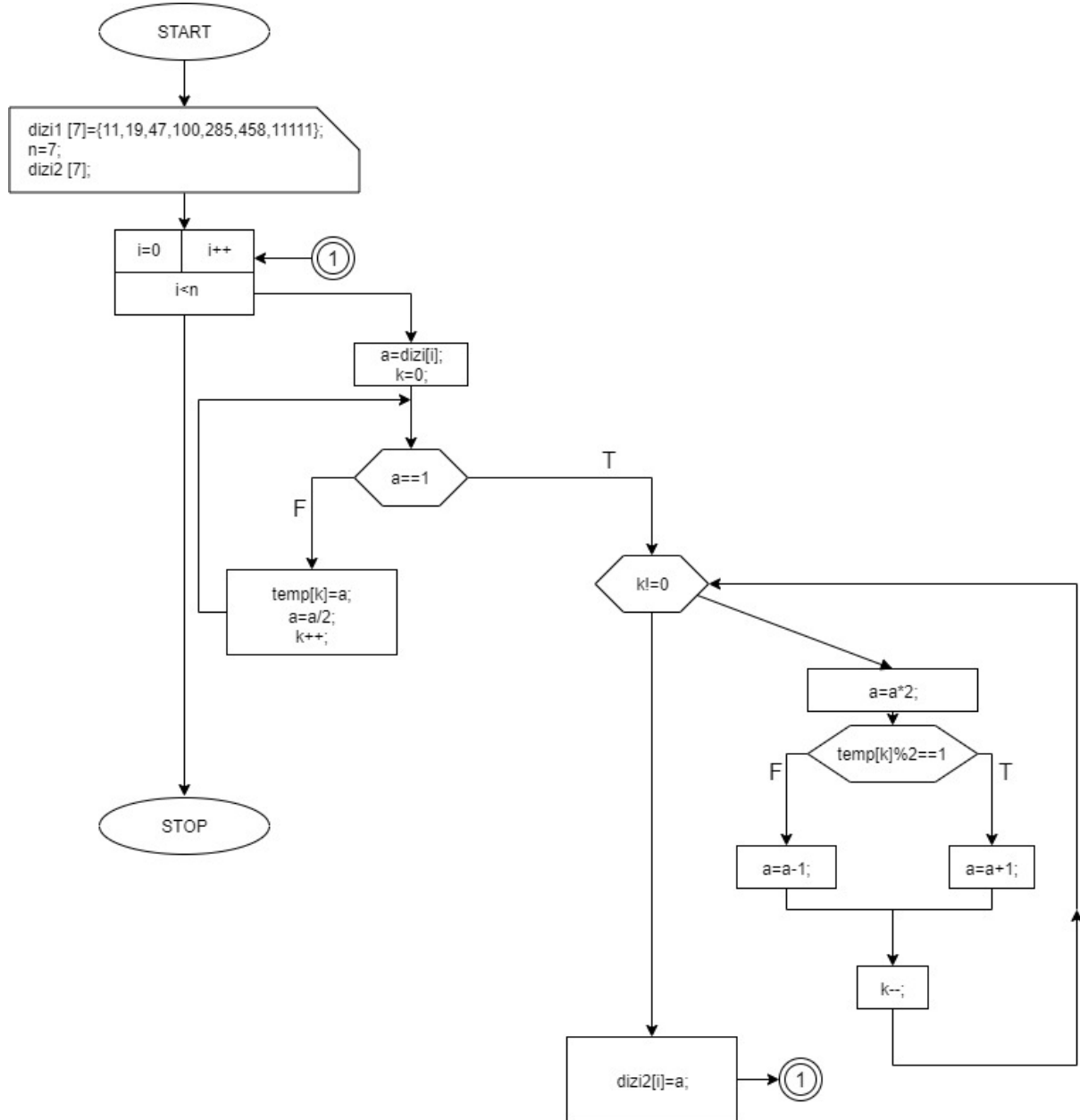
DÖNEM: 2

GRUP NO: 2

DERSİN EĞİTMENİ: Arş.Gör. Furkan ÇAKMAK

Ödev 2:

A) Ana programın ve rekürsif fonksiyonun üst seviye argümanlarla akış diyagramı



B-) Com tipi uygulamalarda stack segment olmadığından değişkenler kod segmentinin bulunduğu segmentin sonundan geriye doğru kaydedilir.

```
C:\>debug 18011028.com
-d cs:FFC0
075A:FFC0 2A 00 3A 12 54 04 E7 00-35 00 42 12 58 04 E7 00  *...T...5.B.X...
075A:FFD0 5B 00 4A 12 5C 04 E7 00-5C 00 52 12 60 04 E7 00  [..J.\...\R.\...
075A:FFE0 6B 00 00 00 64 04 E8 00-F8 00 62 12 68 04 E8 00  k...d.....b.h...
075A:FFF0 17 00 6A 12 6C 04 E8 00-36 00 00 00 FF FF 00 00  ..j.l...6.....
```

Kod segmentinin son bölümünün ilk hali

```
-d cs:ffc0
075A:FFC0 2A 00 3A 12 54 04 E7 00-35 00 42 12 58 04 01 00  *...T...5.B.X...
075A:FFD0 E2 FF 25 01 5A 07 A3 01-01 00 E2 FF 36 01 01 00  ..%.Z.....6...
075A:FFE0 02 00 02 00 EC FF 36 01-02 00 05 00 05 00 F6 FF  .....6.....
075A:FFF0 36 01 05 00 0B 00 FF FF-00 00 60 01 0B 00 00 00  6.....
```

SI değişkeni 0 iken yığının en dolu hali

- 1-) Dizideki ilk eleman 0x000B push edildi.
- 2-) Fonksiyon çağırıldığından 0x0160 değeri push edildi.
- 3-) Base pointerın ilk değeri push edildi. 0x0000
- 4-) AX'in ilk değeri push edildi 0xFFFF.
- 5-) AX e 0x000B değeri atanıp push edildi.
- 6-) AX 2 ye bölünüp bulunan 0x0005 değeri push edildi.
- 7-) AX 1 e eşit olmadığından fonksiyon tekrardan çağırıldı ve 0x0136 değeri push edildi.
- 8-) Base pointerın o anki değeri 0x0FF6 değeri push edildi.
- 9-) AX in o anki değeri push edildi. 0x0005
- 10-) AX 1 e eşit olmadığından AX in o anki değeri yeniden push edildi. 0x0005

```

-d cs:ffc0
075A:FFC0 2A 00 3A 12 54 04 E7 00 35 00 12 12 58 04 01 00 *...T...5.B.X...
075A:FFD0 E2 FF 25 01 5A 07 A3 01 01 00 E2 FF 36 01 01 00 ..%.Z.....6...
075A:FFE0 02 00 02 00 EC FF 36 01 02 00 05 00 05 00 F6 FF .....6.....
075A:FFF0 36 01 05 00 0B 00 FF FF 00 00 60 01 0B 00 00 00 6.....

```

11-) AX 2 ye bölünüp bulunan 2 değeri push edildi. 0x0002

12-) AX 1 e eşit olmadığından fonksiyon tekrardan çağırıldı ve 0x0136 değeri push edildi.

13-) Base pointerın o anki değeri 0x0FEC değeri push edildi.

14-) AX in o anki değeri push edildi. 0x0002

15-) AX 1 e eşit olmadığından AX in o anki değeri yeniden push edildi. 0x0002

16-) AX 2 ye bölünüp bulunan 1 değeri push edildi. 0x0001

17-) 1'e eşit olma kontrolü 2'ye bölme işleminden önce olduğundan fonksiyon tekrardan çağırılıp 0x0136 değeri push edildi.

18-) Base pointerın o anki değeri 0x0FE2 değeri push edildi.

19-) AX in o anki değeri push edildi. 0x0001

Bu işlemden sonra AX 1'e eşit olduğu için yığın bölümü boşalmaya başladı. Bu nedenle SI:0 iken en dolu halin bu olduğu belirlendi.

C) Değişkenlerin Son Hali

dizi1={11, 19, 47, 100, 285, 458, 11111}

n (Eleman sayısı)=7

dizi2={7, 7, 31, 73, 59, 405, 5839}