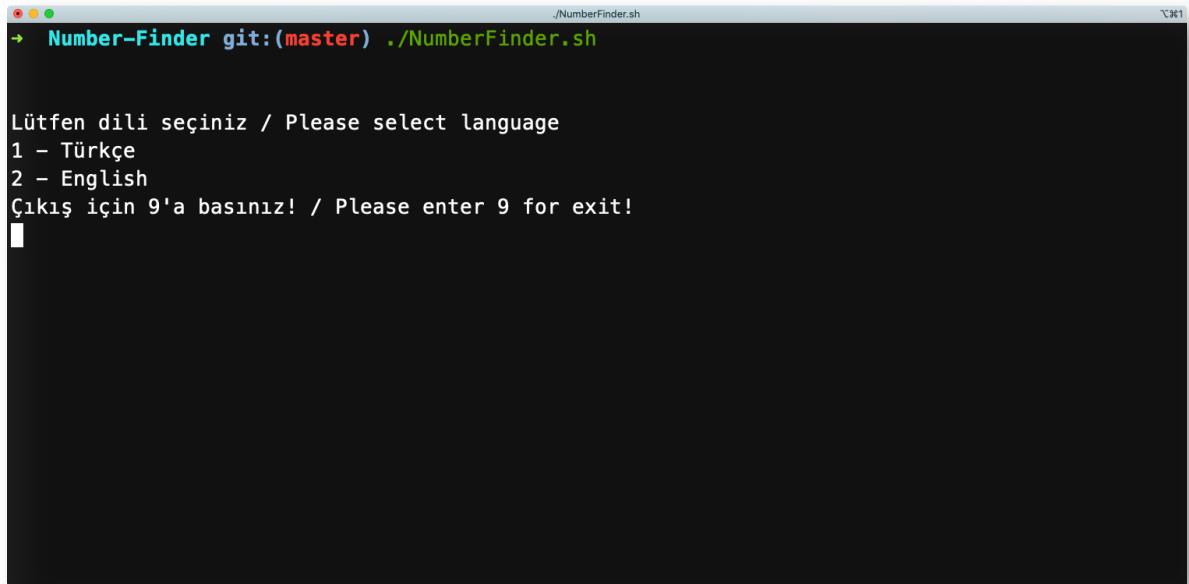


Number Finder

Anıl ÖzTÜRK

(<https://github.com/baykatre/Number-Finder>)



```
Number-Finder git:(master) ./NumberFinder.sh

Lütfen dili seçiniz / Please select language
1 - Türkçe
2 - English
Çıkış için 9'a basınız! / Please enter 9 for exit!
```

BIM404 - Sistem Programlama (System Programming)

Number Finder

Özet

Shell Programlama kullanılarak Türkçe ve İngilizce için sayıdan yazı ve yazıldan sayı dönüşümü yapabilen, ayrıca yazı olarak verilmiş iki sayı üzerinde toplama ve çıkarma işlemlerinin yapılabildiği bir Shell betiği.

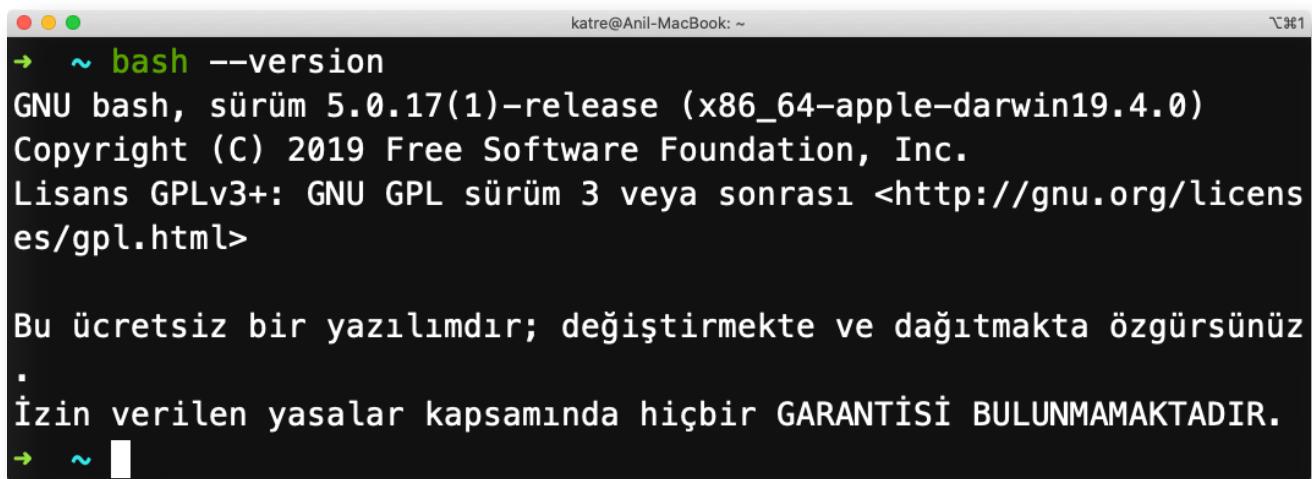
Abstract

A Shell script that can convert numbers from text and text to Turkish and English using Shell Programming, plus addition and subtraction operations on two numbers given as text.

Ortam / Environment

OS: macOS Catalina 10.15.4

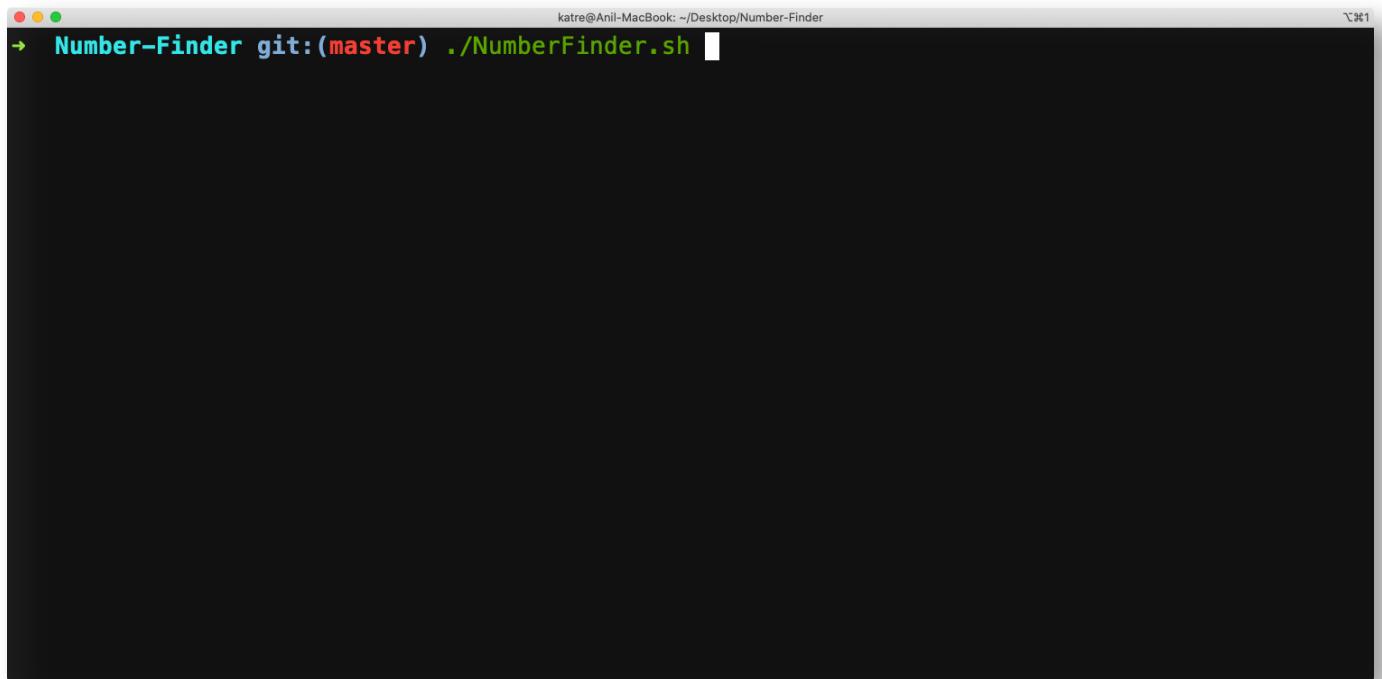
Bash: GNU bash, sürüm 5.0.17(1)-release (x86_64-apple-darwin19.4.0)



```
katre@Anil-MacBook: ~
→ ~ bash --version
GNU bash, sürüm 5.0.17(1)-release (x86_64-apple-darwin19.4.0)
Copyright (C) 2019 Free Software Foundation, Inc.
Lisans GPLv3+: GNU GPL sürüm 3 veya sonrası <http://gnu.org/licenses/gpl.html>

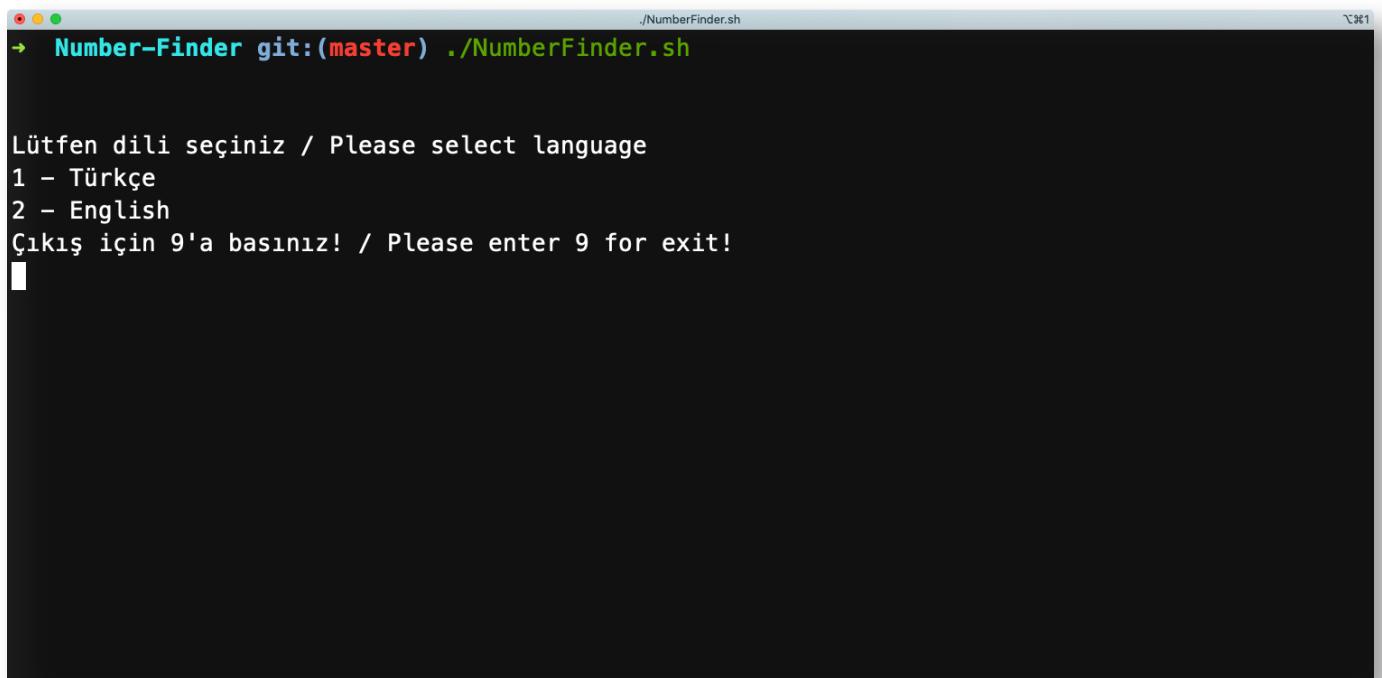
Bu ücretsiz bir yazılımdır; değiştirmekte ve dağıtmakta özgürsünüz
.
İzin verilen yasalar kapsamında hiçbir GARANTİSİ BULUNMAMAKTADIR.
→ ~ █
```

1) Başlat / Start



```
katre@Anil-MacBook: ~/Desktop/Number-Finder
→ Number-Finder git:(master) ./NumberFinder.sh
```

2) Dil seçimi / Language select



```
./NumberFinder.sh
→ Number-Finder git:(master) ./NumberFinder.sh

Lütfen dili seçiniz / Please select language
1 - Türkçe
2 - English
Çıkış için 9'a basınız! / Please enter 9 for exit!
```

3) Seçenekler / Options

```
./NumberFinder.sh
→ Number-Finder git:(master) ./NumberFinder.sh

Lütfen dili seçiniz / Please select language
1 - Türkçe
2 - English
Çıkış için 9'a basınız! / Please enter 9 for exit!
1
1 - Sayı --> Yazı:
2 - Yazı --> Sayı:
3 - Yazı ile işlem:
█
```

Türkçe

```
./NumberFinder.sh
→ Number-Finder git:(master) ./NumberFinder.sh

Lütfen dili seçiniz / Please select language
1 - Türkçe
2 - English
Çıkış için 9'a basınız! / Please enter 9 for exit!
1
1 - Sayı --> Yazı:
2 - Yazı --> Sayı:
3 - Yazı ile işlem:
█
```

English

4) Sayıdan yazıya / From number to text

```
./NumberFinder.sh
→ Number-Finder git:(master) ./NumberFinder.sh

Lütfen dili seçiniz / Please select language
1 - Türkçe
2 - English
Çıkış için 9'a basınız! / Please enter 9 for exit!
1
1 - Sayı --> Yazı:
2 - Yazı --> Sayı:
3 - Yazı ile işlem:
1
Lütfen yazıya çevirilecek sayıyı giriniz. (Max trilyona kadar, trilyon dahil)
450030
Cevap:dort yüz üç bin otuz
```

Türkçe

```
./NumberFinder.sh
→ Number-Finder git:(master) ./NumberFinder.sh

Lütfen dili seçiniz / Please select language
1 - Türkçe
2 - English
Çıkış için 9'a basınız! / Please enter 9 for exit!
2
1 - Number --> Text:
2 - Text --> Number:
3 - Process with text:
1
Please a number for translate to text. (Max to trillion, include trillion)
450324
Solution:four hundred fifty thousand three hundred twenty four
```

English

5) Yazıdan sayıya / From text to number

```
./NumberFinder.sh
→ Number-Finder git:(master) ./NumberFinder.sh

Lütfen dili seçiniz / Please select language
1 - Türkçe
2 - English
Çıkış için 9'a basınız! / Please enter 9 for exit!
1
1 - Sayı --> Yazı:
2 - Yazı --> Sayı:
3 - Yazı ile işlem:
2
Lütfen sayıya çevirilecek yazı karşılığını giriniz. (Max trilyona kadar, trilyon dahil)
dokuz milyar otuz milyon bin beş yüz iki

Cevap: 9030001502
```

Türkçe

```
./NumberFinder.sh
→ Number-Finder git:(master) ./NumberFinder.sh

Lütfen dili seçiniz / Please select language
1 - Türkçe
2 - English
Çıkış için 9'a basınız! / Please enter 9 for exit!
2
1 - Number --> Text:
2 - Text --> Number:
3 - Process with text:
2
Please a text for translate to number. (Max to trillion, include trillion)
one hundred million twenty three thousand three hundred nine
100023309
```

English

6) İşlemler / Operations

```
katre@Anil-MacBook: ~/Desktop/Number-Finder
→ Number-Finder git:(master) ✘ ./NumberFinder.sh

Lütfen dili seçiniz / Please select language
1 - Türkçe
2 - English
Çıkış için 9'a basınız! / Please enter 9 for exit!
1
1 - Sayı --> Yazı:
2 - Yazı --> Sayı:
3 - Yazı ile işlem:
3
1 - Toplama:
2 - Çıkarma:
1
Lütfen operand1 giriniz. (Max milyona kadar, milyon dahil)
bin dort yuz iki

bin dort yuz iki
Lütfen operand2 giriniz. (Max milyona kadar, milyon dahil)
seksen bin otuz

seksen bin otuz
Cevap: 1402 + 80030 = 81432 ( sekzen bir bin dort yuz otuz iki )
→ Number-Finder git:(master) ✘
```

Türkçe

```
katre@Anil-MacBook: ~/Desktop/Number-Finder
→ Number-Finder git:(master) ✘ ./NumberFinder.sh

Lütfen dili seçiniz / Please select language
1 - Türkçe
2 - English
Çıkış için 9'a basınız! / Please enter 9 for exit!
2
1 - Number --> Text:
2 - Text --> Number:
3 - Process with text:
3
1 - Addition:
2 - Subtraction:
1
Please enter operand1. (Max to million, include million)
one thousand fifty three

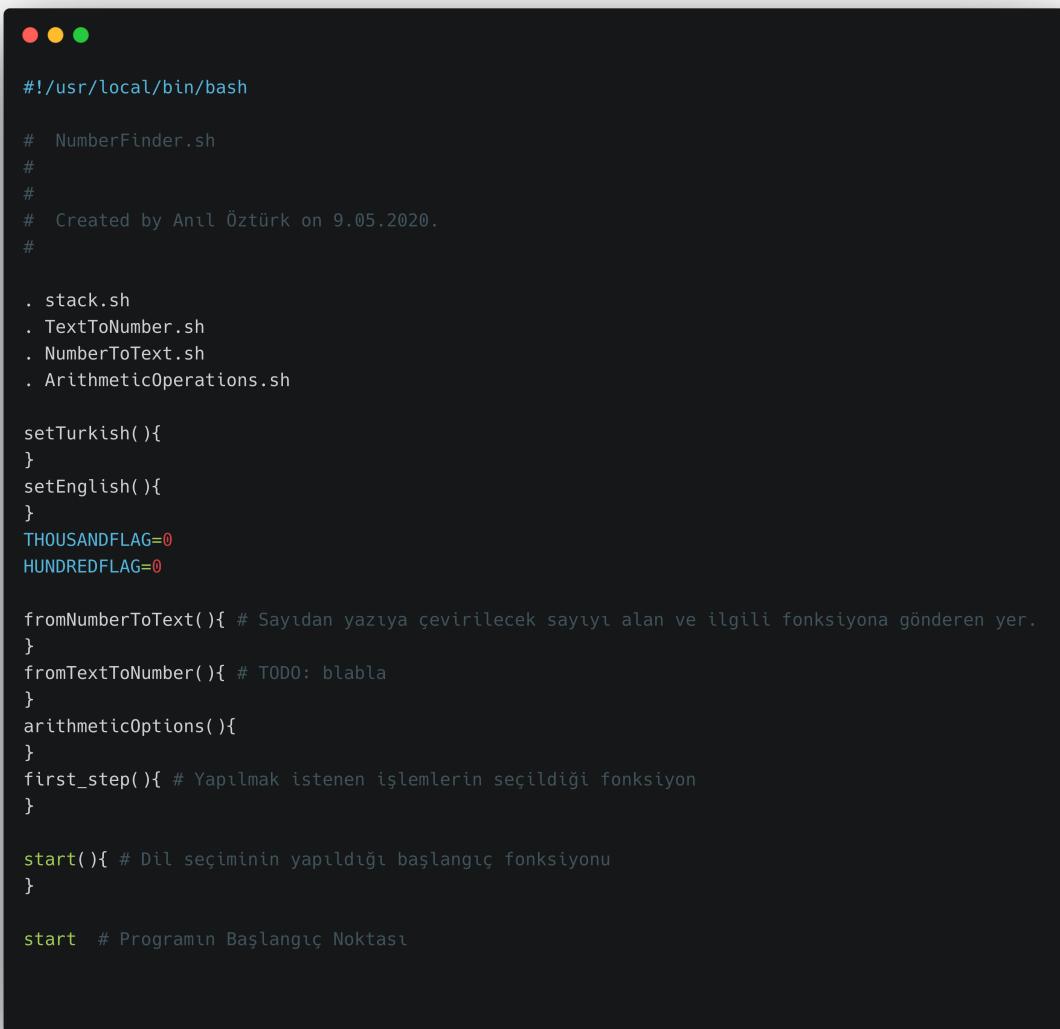
one thousand fifty three
Please enter operand2. (Max to million, include million)
ninety seven
ninety seven
1053 + 97 = 1150 ( one thousand hundred fifty )
```

English

Projeye Giriş / Introduction to The Project

- * Buradaki anlatımlar sadece Türkçe olacaktır. - The explanations here will be in Turkish only.

NumberFinder.sh



```
#!/usr/local/bin/bash

# NumberFinder.sh
#
#
# Created by Anıl Öztürk on 9.05.2020.
#

. stack.sh
. TextToNumber.sh
. NumberToText.sh
. ArithmeticOperations.sh

setTurkish(){}
setEnglish(){}
THOUSANDFLAG=0
HUNDREDFLAG=0

fromNumberToText(){ # Sayıdan yazıya çevirilecek sayılı alan ve ilgili fonksiyona gönderen yer. }
fromTextToNumber(){ # TODO: blabla
}
arithmeticOptions(){}
first_step(){ # Yapılmak istenen işlemlerin seçildiği fonksiyon
}

start(){ # Dil seçiminin yapıldığı başlangıç fonksiyonu
}

start # Programın Başlangıç Noktası
```

- Uygulamanın başlangıç yeri olan NumberFinder.sh dosyası gerekli diğer scriptleri ve fonksiyonları import ettikten sonra en son satırındaki start fonksiyonunu çağrıarak çalışmaya başlar.

NumberFinder.sh - setEnglish() {}

```
...
setEnglish(){
    ATTACHS=( " hundred " thousand " million " billion " trillion ")
    DECIMALS=( " ten " twenty " thirty " fourty " fifty " sixty " seventy " eigthy " ninety ")
    NUMBERS=( " one " two " three " four " five " six " seven " eight " nine ")
    ZERO="zero"
    first_step_text_1="1 - Number --> Text:"
    first_step_text_2="2 - Text --> Number:"
    first_step_text_3="3 - Process with text:"
    first_step_err_msg="Please enter correct choise!"
    arithmetic_msg_1="1 - Addition:"
    arithmetic_msg_2="2 - Subtraction:"
    arithmetic_operand1="Please enter operand1. (Max to million, include million)"
    arithmetic_operand2="Please enter operand2. (Max to million, include million)"
    from_number_to_text_text1="Please a number for translate to text. (Max to trillion, include trillion)"
    from_text_to_number_text1="Please a text for translate to number. (Max to trillion, include trillion)"
    solution="Solution: "
}

...
...
```

- Program çalışınca ilk olarak dil seçimi için ilgili menü kullanıcıya sorulur ve burdaki seçime göre uygulamanın içinde kullanılacak tüm string'ler ilgili dile göre atanır.

NumberFinder.sh - first_step() {}

```
...
first_step(){ # Yapılmak istenen işlemlerin seçildiği fonksiyon
    echo $first_step_text_1
    echo $first_step_text_2
    echo $first_step_text_3
    read -n 1 process
    echo ""
    if [ $process -eq 1 ]
    then
        fromNumberToText
    elif [ $process -eq 2 ]
    then
        fromTextToNumber
    elif [ $process -eq 3 ]
    then
        arithmeticOptions
    else
        echo $first_step_err_msg
        start
    fi
}
...
```

- Start fonksiyonunu dil seçimini yaptıktan sonra bizi yapılacak işlemin yapılacaksı seçim ekranına, yani **first_step** fonksiyonuna, gönderir. Burada da uygulamanın sahip olduğu üç özellik arasından seçim yapıp ilgili fonksiyonlara yönlendirilme gerçekleştirilir.

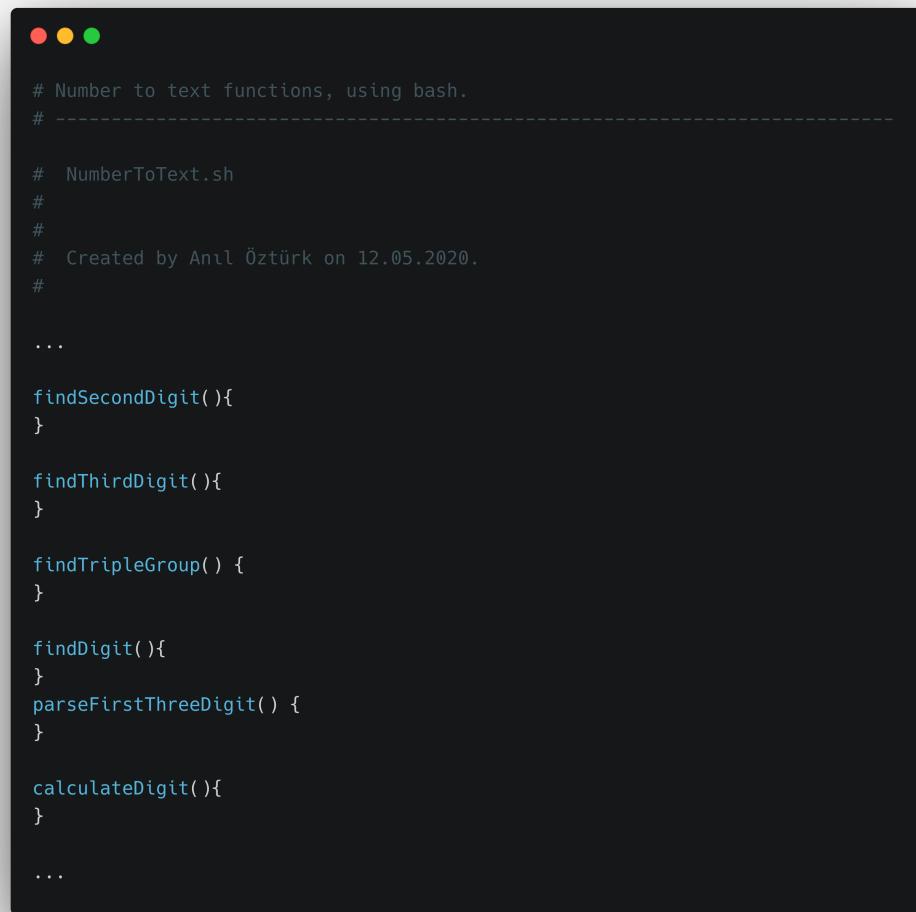
NumberFinder.sh - Yönlendirme Fonksiyonları

```
● ● ●

...
fromNumberToText(){ # Sayıdan yazıya çevirilecek sayıyı alan ve ilgili fonksiyona gönderen yer.
    echo $from_number_to_text_text1
    read -p "" number
    echo -n $solution
    calculateDigit $number
    start
}
fromTextToNumber(){ # TODO: blabla
    echo $from_text_to_number_text1
    read -p "" number
    calculateDigitWithText "$number"
    start
}
arithmeticOptions(){
    echo $arithmetic_msg_1
    echo $arithmetic_msg_2
    read -n 1 process
    echo ""
    if [ $process -eq 1 ]
    then
        startOperations 1
    elif [ $process -eq 2 ]
    then
        startOperations 2
    else
        echo $first_step_err_msg
        start
    fi
}
...
...
```

- Burada yapılan işlem ise seçilen işleme göre kullanıcıdan uygun inputu alıp hesaplama yapılması için ilgili fonksiyonlara göndermek olacak.

NumberToText.sh - Fonksiyonlar



```
# Number to text functions, using bash.
# -----
#
#   NumberToText.sh
#
#
#   Created by Anıl Öztürk on 12.05.2020.
#
#
...
findSecondDigit(){}
findThirdDigit(){}
findTripleGroup() {
}
findDigit(){}
parseFirstThreeDigit() {
}
calculateDigit(){}
...
...
```

- Yazıya çevirilecek olan sayımız ilk olarak **calculateDigit** fonksiyonuna girer ve buradan önce **findDigit** fonksiyonuna giderek basamak sayısını tespit edilir.
- En üstteki **findSecondDigit** ve **findThirdDigit** fonksiyonlarımız gelen sayının ikinci ve üçüncü basamaklarını tespit ediyorlar.
- **findTripleGroup** ise kendisine gönderilen üç basamaklı parçaları bulmaya yarıyor. İlk başta bu fonksiyon için farklı durumlar düşünmüştüm fakat sonrasında gerek kalmadığı için sadece **findThirdDigit** fonksiyonuna gelen parametreyi döndüren gereksiz bir fonksiyona dönüştü.
- Asıl gelen sayıyı parçalayıp onun ilk üç hanesini bulacak olan fonksiyon ise **parseFirstThreeDigit** fonksiyonu. Bu fonksiyon sayının ağırlık değişkenini alarak gerekli matematiksel işlemleri yaparak ilk üç hanesini geri döndürüyor. (Ağırlık = Milyonda 10^6 , Milyarda 10^9 gibi)
- **calculateDigit** fonksiyonu en son sayıyı basamak numarasına göre parse fonksiyonuna gönderip gereken sonucu hesaplayıp döndürüyor.

TextToNumber.sh - Fonksiyonlar



```
# Text to number functions, using bash.
# -----
#
# TextToNumber.sh
#
#
# Created by Anıl Öztürk on 13.05.2020.
#
#
...
findNumbersForText(){}
findDecimalsForText(){}
findHundredForText(){}
findTripleLabelForText(){}
detectAttachs(){}
findDecimals(){}
findTripleGroupFromText(){}
findDigitFromText(){}
calculateDigitWithText(){}
...
...
```

- Buradaki yaklaşımımızı kısaca özetlemek gerekirse öncelikle gelen text'i parçalayıp baştan sonra doğru bir stack'e atıyoruz. Sonra stack'ten teker teker çekip gerekli kontrolleri yaparak sayımızı oluşturuyoruz.
- İlk önce `calculateDigitWithText` fonksiyonuna gelen text `findDigitFromText` fonksiyonuna geçer.
- Burada, `findDigitFromText`, yeni bir stack oluşturulup text'i parçalayıp içine atıyoruz bu stack'in. Sonrasında yine üçlü grupları bulmak üzere `findTripleGroupFromText` fonksiyonunu çağırıyoruz.
- Burada stack uzunluğu tespit edildikten sonra bir döngü ile her elemanın üzerinden geçip ön eklerin (attach) tespiti için `detectAttachs` fonksiyonuna gönderiyoruz.

- `detectAttach` fonksiyonunun asıl işlevi stack'ten çıkan kelimelerin içinde “bin, milyon, milyar ve trilyon” eklerini aramak aslında. Bu kelimelerin varlığında onları ilgili fonksiyona özel parametrelerle yollayıp rekürsif olarak hesaplanması sağlıyor. Eğer bu ön ekleri bulamazsa yüzdelik, onluk ve birlik hesaplamalarını yapan `findDecimals` fonksiyonuna gönderiyor ilgili kelimeleri.
 - `findTripleLabelForText` fonksiyonu “bin, milyon, milyar ve trilyon” kelimeleri geldiğinde kendini tekrar çağırarak alt seviyedeki basamaklarının hesaplamalarını da içinde yapabilecek şekilde tasarlandı.
- * Burada bana ciddi problem yaratan kısım Türkçe’de 1000 sayısı telaffuz edilirken “bir bin” şeklinde değil de sadece “bin” şeklinde telaffuz edildiği için benim “bin” kelimesi gördüğüm anda bir ekleyip eklememem gibi karışık bir durumun içine girmiştim. Sonradan bunu bir ön ek gördüğümde onunla diğer ön ek arasındaki tüm sayıları tespit ederek çözebildim. Sancılı oldu ama oldu. :)
- `findNumbersForText`, `findDecimalsForText` ve `findHundredForText` fonksiyonları ise sırasıyla birler, onlar ve yüzler basamakları için gerekli taramaları yapan fonksiyonlar diyebiliriz.

ArithmeticOperations.sh - Fonksiyonlar

```
# Functions for arithmetic operations, using bash.  
# -----  
  
# ArithmeticOperations.sh  
#  
#  
# Created by Anıl Öztürk on 15.05.2020.  
#  
  
...  
  
getOperand( ){  
}  
  
operations( ){  
}  
  
addition( ){  
}  
  
subtraction( ){  
}  
  
startOperations( ){  
}  
  
...
```

- Buraya kullanıcının seçmiş olduğu işlem bilgisi **startOperations** fonksiyonuna gelip burada **operations** fonksiyonuna geçer.
- **operations** fonksiyonu ise kullanıcıdan gerekli operandların alınması için **getOperand** fonksiyonunu çağırıp iki operandı da buradan alıdır.
- Buradan alırken de basamak kontrolü için ilgili fonksiyonları çağırıp sayının milyon kontrolünü yapar.
- Eğer sorun yoksa **addition** ve **subtraction** fonksiyonlarına yönlendirmeler yapılmış gerekli işlem gerçekleştirilir.

Dikkat edilmesi gerekenler

```
jNumberFinder.sh
→ Number-Finder git:(master) ✘ ./NumberFinder.sh

Lütfen dili seçiniz / Please select language
1 - Türkçe
2 - English
Çıkış için 9'a basınız! / Please enter 9 for exit!
1
1 - Sayı --> Yazı:
2 - Yazı --> Sayı:
3 - Yazı ile işlem:
2
Lütfen sayıya çevirilecek yazı karşılığını giriniz. (Max trilyona kadar, trilyon dahil)
on bes
stack.sh: line 18: declare: -g: invalid option
declare: usage: declare [-afFirtx] [-p] [name[=value] ...]
stack.sh: line 19: declare: -g: invalid option
declare: usage: declare [-afFirtx] [-p] [name[=value] ...]
Cevap: 15
```

- Uygulama geliştirilirken stack işlemleri için bazı parametreler kullanmamız gerekiyordu. Bunu uygulamaya implement ettiğimizde mevcut bash sürümümüzle doğru çalışmasına rağmen warning hataları alıyordu. Bu yüzden güncel bash sürümünü indirip “/usr/local/bin/bash” dizininden “GNU bash, sürüm 5.0.17(1)-release (x86_64-apple-darwin19.4.0)” sürüm etiketli sürümü kullanıyoruz.
- Yukarıdaki uyarıları aldığımız bash sürümü ise “GNU bash, version 3.2.57(1)-release (x86_64-apple-darwin19)”

```
katre@Anil-MacBook: ~/Desktop/Number-Finder
→ Number-Finder git:(master) ✘ /bin/bash --version
GNU bash, version 3.2.57(1)-release (x86_64-apple-darwin19)
Copyright (C) 2007 Free Software Foundation, Inc.
```

Anıl Öztürk
30116010

Teşekkürler...