Strings in Go

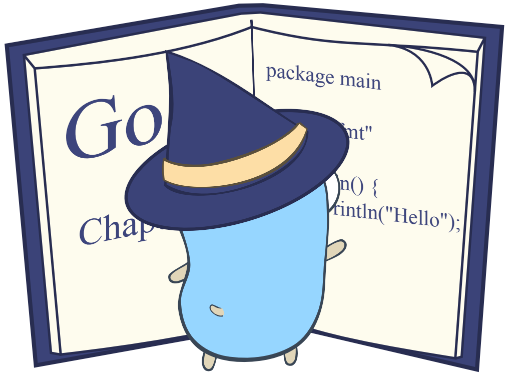
Why can we fmt.Println("Hello, 世界")? Quick overview of runes, bytes, unicode utf-8

22 May 2019

Arkadiusz Noster
Warsaw University of Technology computer science student

What you'll learn

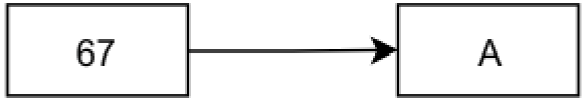
- What ASCII, unicode and UTF-8 are
- Encoding of Go source, strings and some tweaks with it



Gopher by Renée French (https://github.com/egonelbre/gophers)

Text encodings standards

character encoding - the mapping of numbers to glyphs (character, symbol).



Code space - a range of numerical values available for encoding characters

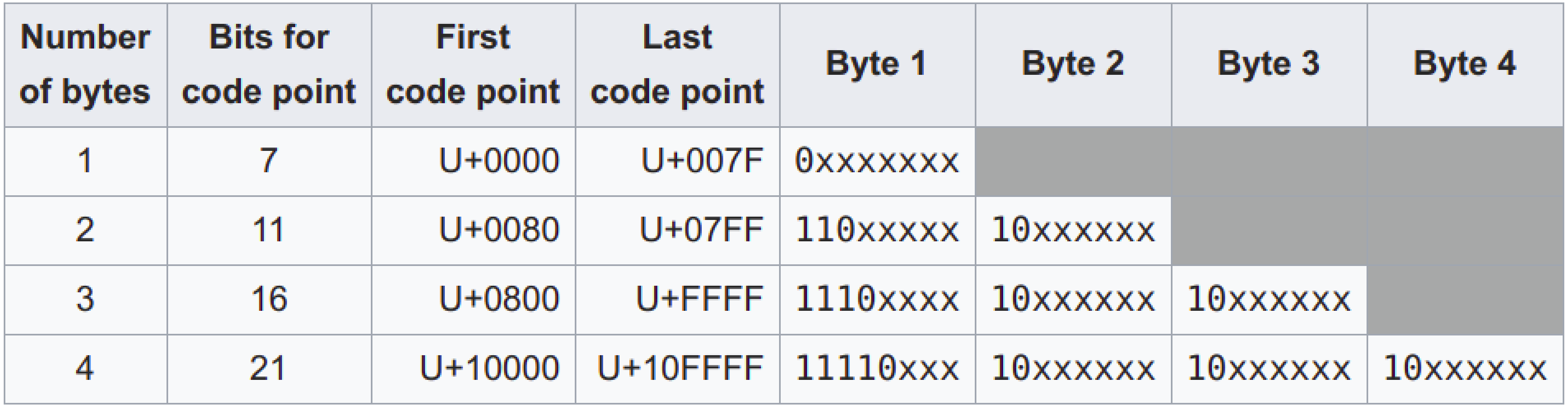
Code point - a number within the codespace

Character Encoding Scheme - the way we represent code point

Unicode

unicode-table.com/en/ (https://unicode-table.com/en/)

utf-8



Number of bytes	Bits for code point	First code point	Last code point	Byte 1	Byte 2	Byte 3	Byte 4
1	7	U+0000	U+007F	0xxxxxxx			
2	11	U+0080	U+07FF	110xxxxx	10xxxxxx		
3	16	U+0800	U+FFFF	1110xxxx	10xxxxxx	10xxxxxx	
4	21	U+10000	U+10FFFF	11110xxx	10xxxxxx	10xxxxxx	10xxxxxx

from wikipedia (https://en.wikipedia.org/wiki/UTF-8#Description)

Golang source code must be utf-8 encoded!

```go
package main

import "fmt"

func main() {
    Überraschung := "Die Variablen können deutsche Namen haben"
    fmt.Println(Überraschung)
    переменная := "У переменных тоже могут быть русские имена!😄"
    fmt.Println(переменная)
}
```

Run

But

```
Identifiers name program entities such as variables and types.
An identifier is a sequence of one or more letters and digits.
```

So no emojis in variable names 😭

Strings

```
package main

import (
    "fmt"
    "reflect"
)

func main(){
    var str string =  "Hello"
    fmt.Println("string's element type:",reflect.TypeOf(str[0]))
    ruStr := "привет"
    fmt.Printf("% x\n", str)
    fmt.Printf("% x\n", ruStr)
    var b []byte = []byte(ruStr)
    fmt.Println(b)
    ruStr = string(b)
    fmt.Println(ruStr)
    //str[0] = 0x01;



}
```

Run

Literals

```
package main

func main() {
    println('b' == 98)
    println('b' == '\142')
    println('b' == '\x62')
    println('b' == '\u0062')
    println('b' == '\U00000062')
}
```

Run

Literals #2

```go
package main

import "fmt"

func main(){
    const Hello = "Hello, Мир"
    type myType string
    var k myType = Hello
    fmt.Println(k)

    // WYSIWYG literals - backquotes!
    backQuoteStr := `This string
    will have
    tabs in it`
    fmt.Println(backQuoteStr)
}
```

Run

Thank you

Arkadiusz Noster
Warsaw University of Technology computer science student
arkadiusz.noster@gmail.com (mailto:arkadiusz.noster@gmail.com)
https://www.linkedin.com/in/arkadiusz-noster/ (https://www.linkedin.com/in/arkadiusz-noster/)