Participation 10

Exercise 10.1

```
fibonacci <- numeric(10)
fibonacci[1] <- 1
fibonacci[2] <- 1
for (i in 3:10){
    fibonacci[i] <- fibonacci[i-2] + fibonacci[i-1]
 }
 fibonacci
```

```
##  [1]  1  1  2  3  5  8 13 21 34 55
```

Exercise 10.2

```
fibonacci <- function(len){
    if (len == 1){return(1)}
    else if (len == 2) {return(c(1,1))}
    else{
        fib <- numeric(len)
        fib[1] <- 1
        fib[2] <- 1
        for (i in 3:len){fib[i] <- fib[i-2] + fib[i-1]}
        return(fib)
    }

}
fibonacci(20)
```

```
##  [1]    1    1    2    3    5    8   13   21   34   55   89  144  233  377  610
## [16]  987 1597 2584 4181 6765
```

Exercise 10.3

```
in.between <- function(x){
    if( (x <= 1) & (x >= 0) ){
        return(1)
    }
    return(0)
}
```

Exercise 10.4

```
sapply(c(-9,-0.1,0,0.1,1),in.between)
```

```
## [1] 0 0 1 1 1
```

Exercise 10.5

```
head(letters,10)
```

```
##  [1] "a" "b" "c" "d" "e" "f" "g" "h" "i" "j"
```

```
tail(LETTERS,10)
```

```
##  [1] "Q" "R" "S" "T" "U" "V" "W" "X" "Y" "Z"
```

```
LETTERS[22:24]
```

```
## [1] "V" "W" "X"
```

Exercise 10.6

```
for(n in seq(100)){
    if (n %% 3 == 0) {
        if (n %% 5 == 0){
            cat(n," FizzBuzz\n")
        } else {cat(n," Fizz\n")}
    }
    if (n %% 5 == 0 & n %% 3 != 0) {
        cat(n," Buzz\n")
    }
}
```

```
## 3  Fizz
## 5  Buzz
## 6  Fizz
## 9  Fizz
## 10  Buzz
## 12  Fizz
## 15  FizzBuzz
## 18  Fizz
## 20  Buzz
## 21  Fizz
## 24  Fizz
## 25  Buzz
## 27  Fizz
## 30  FizzBuzz
## 33  Fizz
## 35  Buzz
## 36  Fizz
## 39  Fizz
## 40  Buzz
## 42  Fizz
## 45  FizzBuzz
## 48  Fizz
## 50  Buzz
## 51  Fizz
## 54  Fizz
## 55  Buzz
## 57  Fizz
## 60  FizzBuzz
## 63  Fizz
```

```
## 65  Buzz
## 66  Fizz
## 69  Fizz
## 70  Buzz
## 72  Fizz
## 75  FizzBuzz
## 78  Fizz
## 80  Buzz
## 81  Fizz
## 84  Fizz
## 85  Buzz
## 87  Fizz
## 90  FizzBuzz
## 93  Fizz
## 95  Buzz
## 96  Fizz
## 99  Fizz
## 100  Buzz
```

Exercise 10.7

```
string <- "here is an example of an unordinary sentence."
updated.string <- unlist(strsplit(tolower(string)," "))

unique(updated.string)
```

```
## [1] "here"       "is"         "an"         "example"    "of"
## [6] "unordinary" "sentence."
```

```
new.vec <- c(1,2,3,4,5,7,3,2)
unique(new.vec)
```

```
## [1] 1 2 3 4 5 7
```