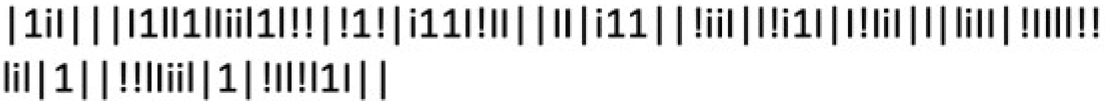
Generating password is a hustle and I don't want to go into the philosophy of passwords. Strong password, lengthy password, automatically generated password, etc. For me, they are all hard to remember, even harder to type (due to typos, different language setups, OS, software, etc.). In return, let have another useless R function, that will generate useless, – hard-to-use and impossible to remember – password, that will comply to all the standard rules (8 characters or more, one upper case, a kidney, a number, a special character, section from war & Peace, etc).

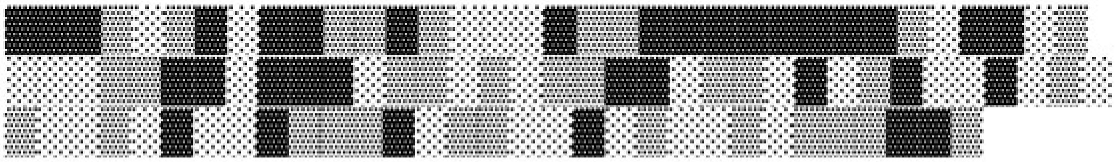
Typical "strong" password that human brain struggles to comprehend.

Xn&qI7k#Ftn0HVA

To make this password generation rather wacky, let me play with following characters:



|1il|||l1ll1lliil1l!!|!1!|i11l!ll||ll|i11||!iil|l!i1l|l!lil|l|lill|!llll!!
lil|1||!!lliil|1|!ll!l1l||



There is less diversity in characters but nonetheless, the correct length and small caps, number, special characters and kidney – all comply with the regular quality assurance.

Wacky R function generates a set of ill generated password that would be even harder to type, and stupic to remember.

```
# Running on Linux/MacOS
WackyPassword <- function(WP_length){
  #charblock1 = c(176:178, 185: 188, 200:206)
  charblock1 <- c("\u2591","\u2592","\u2593")
  charblock2 = c(73,105,108,124,49,33)
  numberblock3 <- sample(0:9, length(5),replace = TRUE)

  pass = ""
  Encoding(pass) <- "UTF-8"
  ran2 <- floor(sample(1:WP_length/2))
  ran1 <- floor(sample(1:WP_length/2))
      while (nchar(pass) <= WP_length) {
        res2 <- sample(charblock2, 100,replace = TRUE)
        res2 <- rawToChar(as.raw(res2))
        Encoding(res2) <- "UTF-8"
        start2 <- sample(1:90,1)
        pass <- paste0(pass,substr(res2,start2,start2+ran2),collapse="", sep=
"")
```

```
        res1 <- sample(charblock1, 100,replace = TRUE)
        Encoding(res1) <- "UTF-8"
        start1 <- sample(20:70,1)
        res <- paste0(res1, sep = "", collapse = "")
        pass <- paste0(pass,substr(res,start1,start1+ran1), sep="", collapse =
"")
          }

  cat(eval(substr(pass,1,WP_length)))
}
```

Running the function is as simple as:

```
WackyPassword(18)
```

but the results can be as useless and wacky as you can imagine 🙂

!il1|l▓░▓▓▓░!l|lli

Good luck typing this and I buy a beer to every community member brave enough to use one