

# Gerenciador de Senhas

Brenno Rodrigues de Carvalho da Silva

Natã dos Santos Carvalho

11 de maio de 2017

## 1 Introdução

Nesta etapa do trabalho consiste na implementação das funções de leitura de arquivo, tratamento e classificação das senhas, em `Perl` utilizando módulos e subrotinas.

## 2 Implementação

As funções de tratamento e classificação das senhas são executadas recebendo parâmetros como nome, data de nascimento e senha do usuário. Para a classificação das senhas foram adotados os seguintes critérios:

- Utilização de caracteres numéricos - A senha recebe um 1 ponto pela utilização deste critério, caso contrário não recebe nada;
- Utilização de caracteres alfabéticos minúsculos - A senha recebe um 1 ponto pela utilização deste critério, caso contrário não recebe nada;
- Utilização de caracteres alfabéticos maiúsculos - A senha recebe um 1 ponto pela utilização deste critério, caso contrário não recebe nada;
- Utilização de caracteres especiais - A senha recebe um 2 ponto pela utilização deste critério, caso contrário não recebe nada;

Se o somatório dos pontos recebidos nestes 4 critérios for um total de 5 pontos, a senha é classificada como forte. Para um total de 2 ou menos pontos a senha é classificada como fraca e é recomendado a utilização dos critérios acima.

Quando o total de pontos for igual a 3 ou 4 pontos, o programa entra em uma subrotina que verifica a ocorrência do nome e da data de nascimento na composição da senha, se houver pelo menos uma ocorrência destas, a senha é classificada como fraca e é recomendado retirar o nome e/ou a data de nascimento da mesma.

Caso ambas as verificações citadas anteriormente sejam falsas, a senha é classificada como média e deixamos a critério do usuário melhorá-la ou não. Foram implementadas as seguintes sub rotinas:

## 2.1 NumericCharacters

Esta subrotina que detecta a incidência de um ou mais caracteres numéricos. Caso haja este tipo de caracteres, a variável *\$classificacao* recebe 1 pontos. Caso contrário, não recebe nenhuma pontuação.

```
1  sub NumericCharacters{
2      my $classificacao;
3      my @senha = @_ ;
4      if($senha [0] =~ /\d/){
5          #print "Tem numeros\n";
6          $classificacao = 1;
7          return $classificacao;
8      }
```

## 2.2 LowerCaseCharacters

Esta subrotina detecta a incidência de um ou mais caracteres alfabéticos minúsculos. Caso haja este tipo de caracteres, a variável *\$classificacao* recebe 1 ponto. Caso contrário, não recebe nenhuma pontuação.

```
1  sub LowerCaseCharacters{
2      my $classificacao;
3      my @senha = @_ ;
4      if($senha [0] =~ /[a-z]/){
5          # print "Tem letras minusculas\n";
6          return $classificacao = 1;
7      }
8  }
```

## 2.3 UpperCaseCharacters

Esta subrotina detecta a incidência de um ou mais caracteres alfabéticos maiúsculos. Caso haja este tipo de caracteres, a variável *\$classificacao* recebe 1 ponto. Caso contrário, não recebe nenhuma pontuação.

```
1  sub upperCaseCharacters{
2      my $classificacao;
3      my @senha = @_ ;
4      if($senha [0] =~ /[A-Z]/){
5          # print "Tem letras Maisculas\n";
6          return $classificacao = 1;
7      }
8  }
```

## 2.4 SpecialCharacters

Esta subrotina detecta a incidência de um ou mais caracteres especiais. Caso haja este tipo de caracteres, a variável *\$classificacao* recebe 2 pontos. Caso contrário, não recebe nenhuma pontuação.

```

1  sub specialCharacters{
2      my $classificacao;
3      my @senha = @_;
4      if($senha [0] =~ /[!@#\$%\^\~|&*_{+}\/]){
5          # print "Tem special\n";
6          return $classificacao = 2;
7      }
8  }

```

## 2.5 CheckBirthday

Esta subrotina é executada somente se o somatório dos pontos concedidos nas subrotinas anteriores seja igual a 3 ou 4 pontos. Ela detecta a utilização da data de nascimento do usuário na composição da senha. Caso a senha possua a informação citada acima, e retornado para a rotina principal o valor 1 que indicará que a senha deveria receber a classificação de senha fraca.

```

1  sub checkBirthday {
2      my @nascimento = $_[1];
3      #print $nascimento [0];
4      my @senha = $_[0];
5      if (index($senha [0], $nascimento [0]) != -1){
6          #print "Senha utiliza a data de nascimento\n";
7          return 1;
8      }
9  }

```

## 2.6 CheckName

Esta subrotina também é executada somente se o somatório dos pontos concedidos nas subrotinas anteriores seja igual a 3 ou 4 pontos. Ela detecta a utilização do nome do usuário na composição da senha. Caso a senha possua esta informação, é retornado para a rotina principal o valor 1 que indicará que a senha deverá receber a classificação de senha fraca.

```

1  sub checkName{
2      my @nome = $_[1];
3      my @senha = $_[0];
4      my $bool = 0;
5      if (index($senha [0], $nome [0]) != -1){
6          #print "Senha utiliza o nome\n";
7          return $bool=1;
8      }
9  }

```

## 3 Casos de Uso

Estas subrotinas agrupadas em um módulo serão utilizadas pelo programa principal implementado em C++. No momento estamos usando um programa de testes para verificar a correta execução das mesmas.

## 4 Conclusão

Algumas subrotinas deverão ser revisadas posteriormente pois em alguns dos nossos testes elas não se comportaram da maneira que gostaríamos. Estaremos providenciando todas as melhorias necessárias para cumprir com o que foi proposto na primeira etapa do trabalho.