 <small>ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO</small>	Tipo de Prova Exame – Parte 1	Ano letivo 2019/2020	Data 14/02/2020
	Curso Licenciatura em Engenharia Informática Licenciatura em Segurança Informática em Redes de Computadores	Hora 10:00	
	Unidade Curricular Fundamentos de Programação	Duração 2:00 horas	

- b) Implemente a função **inverte**, que imprime na consola a matriz com as cores invertidas (3V).
- c) Implemente a função **altera**, que altera a intensidade da cor da imagem recebendo a matriz e um valor percentual (entre 0 e 1). O valor pode ser positivo e negativo indicando deste modo se deve reduzir ou aumentar a intensidade da cor (3V).

3. Implemente a função **ltrim** (do inglês *left trim*). Esta função remove todos os espaços iniciais de uma string **str** dada como parâmetro. Por exemplo, se a string inicial for “___programar” (em que, por clareza, o carácter _ representa um espaço), o resultado deve ser “programar”. A função deve devolver o número de espaços removidos.

(3.5V)


(20 min)

4. Implemente a função **unzip_string**. Esta função recebe uma string **in** que contém uma mistura de letras e números (entre 2 e 9), e expande a string **in** repetindo *n* vezes cada letra que seja seguida de um número *n*. O resultado deve ser colocado na string **out**. Assuma que **out** tem tamanho suficiente para acomodar a nova string. Por exemplo, para **in** = “AB3CA2” o conteúdo final de **out** deverá ser “ABBBCAA”. Assuma que a string **in** está sempre bem formada. Por exemplo: não começa com um número, não tem dois números consecutivos, etc.

(3.5V)

(20 min)

Recorda-se ainda que o carácter 0 (zero) na tabela ascii tem o valor decimal 48.

 ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO	Tipo de Prova Exame – Parte 1	Ano letivo 2019/2020	Data 14/02/2020
	Curso Licenciatura em Engenharia Informática Licenciatura em Segurança Informática em Redes de Computadores	Hora 10:00	
	Unidade Curricular Fundamentos de Programação	Duração 2:00 horas	

Anexo

Conteúdo do header file API_Leitura.h:

```
void readShort(short *const value, const short minValue,
               const short maxValue, char const* const message);

void readInt(int *const value, const int minValue,
             const int maxValue, char const* const message);

void readLong(long *const value, const long minValue,
              const long maxValue, char const* const message);

void readFloat(float *const value, const float minValue,
               const float maxValue, char const* const message);

void readDouble(double *const value, const double minValue,
                const double maxValue, char const* const message);

void readChar(char *const value, char const* const message);

bool readString(char *const value, const unsigned int size,
                char const* const message);

void readBool(bool *const value, char const* const message);
```

Conteúdo parcial do header file string.h:

```
size_t strlen (const char *s)
char * strcpy (char *restrict to, const char *restrict from)
int strcmp (const char *s1, const char *s2)
```