P.PORTO		Tipo de Prova Exame – Parte 1	Ano letivo 2019/2020	Data 14/02/2020
	ESCOLA Superior	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

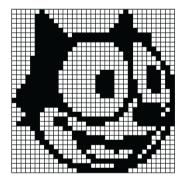
Observações:

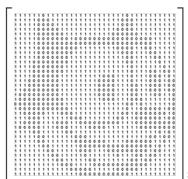
- Pode trocar a ordem das questões, desde que as identifique convenientemente.
- É permitida a utilização da biblioteca de input de dados (API_Leitura.h) abordada nas aulas, disponibilizada em anexo.
- Todos os exercícios devem ser resolvidos utilizando a linguagem de programação C.
- Qualquer tentativa de fraude implica a anulação do exame.
- 1. Crie a função **existe** que recebe duas matrizes quadradas:

(4V)

(20 min)

• **imagem** – matriz (ex. imagem seguinte) contendo apenas os valores 0 e 1 (onde 1 indica que o pixel é branco).





35x35

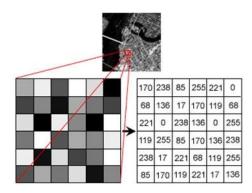
• parte - matriz (também ela com valores 0 e 1).

Esta função deve verificar se a matriz **parte** se encontra em **imagem**. Caso exista, a função deverá devolver o valor 1, caso contrário, o valor 0. O tamanho das matrizes é definido, respetivamente, nas constantes **MAX_IMAGEM** e **MAX_PARTE**.

2. (9V)

Considere um programa que lida com uma matriz quadrada **imagem** contendo valores entre 0 e 255 (indicando a intensidade do branco, exemplo na imagem seguinte) de uma imagem e uma constante **MAX_IMAGEM** que indica o tamanho da matriz.

(60 min)



a) Implemente a função **conta_cores**, que recebe a matriz e que escreve na consola para os valores possíveis de cores a quantidade de pixéis na imagem que têm essa cor (3V).

ESTG-PR05-Mod013V2 Página 1 de3

P.PORTO		Tipo de Prova Exame – Parte 1	Ano letivo 2019/2020	Data 14/02/2020
	ESCOLA SUPERIOR DE TECNOLOGIA E GESTÃO	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. (3.5V)

Implemente a função **Itrim** (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 caracter _ representa um espaço), o resultado deve ser "programar". A função deve devolver o número de espaços removidos.

(20 min)

4.

(3.5V)

(20 min)

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 nvezes 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.

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

ESTG-PR05-Mod013V2 Página 2 de3

P.PORTO		Tipo de Prova Exame – Parte 1	Ano letivo 2019/2020	Data 14/02/2020
	ESCOLA Superior	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)
```

ESTG-PR05-Mod013V2 Página 3 de3