

**INSTITUTO POLITÉCNICO DE lEIRIA**

**Engenharia Informática – Programação Avançada 2.º Ano – 1.º Semestre**



PROJETO PRÁTICO

*paCodec Tool – Etapa 2*

Paulo Penicheiro nº2130628 e por Ruben Bernardes 2130664 declaram sob compromisso de honra que o presente trabalho (código, relatórios e afins) foi integralmente realizado por nós, sendo que as contribuições externas se encontra claramente e inequivocamente identificadas no relatório e no próprio código fonte. Mais se declara que os estudantes acima identificados não disponibilizaram o código ou partes dele a terceiros.

|  |  |  |
| --- | --- | --- |
|  | | |
| PAULO PENICHEIRO |  | RUBEN MIGUEL |
| N 2130628 | ENHENHARIA INFORMÁTICA | N º 2130664 |
|  |  |  |
|  | Prof. PATRÍCIO DOMINGUES |  |
|  | LEIRIA, NOVEMBRO 2015 |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

**ÍNDICE**

[1 Introdução 1](#_Toc438043938)

[2 Funções Implementadas 1](#_Toc438043939)

[Funcionalidades comuns às funções 1](#_Toc438043940)

[Função about 1](#_Toc438043941)

[Função install\_signal\_handler 1](#_Toc438043942)

[Função process\_signal 2](#_Toc438043943)

[Função endecodePGM 2](#_Toc438043944)

[Função parallelEncode 2](#_Toc438043945)

[Função quadError 3](#_Toc438043946)

[Função encodeBlockimgX 3](#_Toc438043947)

[Função process\_block 3](#_Toc438043948)

[Função getNewTask 3](#_Toc438043949)

[Função build\_cod 3](#_Toc438043950)

[Função Write\_Z2\_cod\_to\_file 3](#_Toc438043951)

[Função validate\_dic\_pgm 3](#_Toc438043952)

[Função Write\_Z5\_cod\_to\_file 4](#_Toc438043953)

[Função main 5](#_Toc438043954)

[3 - Conclusão 5](#_Toc438043955)

[4 - Objetivos 6](#_Toc438043956)

[Objetivos Atingidos 6](#_Toc438043957)

[Objetivos Não Atingidos 6](#_Toc438043958)

[5 - Bibliografia 6](#_Toc438043959)

# 1 Introdução

#### O relatório descreve as atividades realizadas no âmbito da execução do trabalho prático do curso de Engenharia Informática na disciplina de Programação Avançada – 2.ª Etapa.

Consiste, na implementação de um utilitário em modo consola chamado paCodec cujo objetivo é permitir efetuar a Codificação/descodificação com perda, de imagens.

Nas páginas seguintes é possível encontrar as opções tomadas na criação do programa, os objetivos cumpridos e não cumpridos nesta 2ª Etapa do trabalho.

# 2 Funções Implementadas

## Funcionalidades comuns às funções

Uma vez que optámos usar a versão base da etapa 1 fornecida pelos professores, mantivemos todas as funções do utilitário paCodec incluídas no ficheiro *utils.c,* estando os respectivos protótipos declarados no ficheiro *utils.h.*

Passamos a descrever essas mesmas funções:

### Função about

Esta função é desencadeada caso seja dado o argumento *--about* (forma longa) ou *–A* (forma abreviada), e mostra a informação dos alunos que desenvolveram o utilitário paCodec.

### Função install\_signal\_handler

Esta função tem como finalidade analisar um sinal de sistema ativado pelo utilizador e atribuir um valor à respetiva *flag*. No caso desta primeira etapa apenas será avaliado o sinal SIGINT e caso este tenha sido ativado a *flag sigint\_flag* ficará com o valor a “1”.

Esta *flag* trata-se de uma varável global irá permitir a qualquer função verificar a ativação deste sinal.

### Função process\_signal

Esta função tem como objetivo processar o sinal ativado pelo utilizador e agir em conformidade fechando também todos os ficheiros abertos e informando o utilizador que a aplicação foi terminada.

## Função endecodePGM

Esta função é desencadeada caso seja dado o argumento *--encode* e é responsável pela codificação de ficheiros de imagem do tipo .pgm transformando-as num ficheiro do tipo .cod, com perdas de qualidade.

A função recebe a estrutura do ficheiro .pgm e a estrutura do respetivo dicionário, mais o nome do ficheiro codificado assim como o nome do respetivo dicionário.

Como algoritmo de descodificação, analisa cada valor do ficheiro .cod, atribuindo à matriz o valor de acordo com o índice do dicionário, preenchendo também o cabeçalho da estrutura do ficheiro descodificado.

## Função parallelEncode

## Função quadError

Esta função vai calcular o erro quadrático da imagem, onde para cada bloco da imagem de entrada é efetuada uma pesquisa no dicionário de padrões por forma a ser encontrado o bloco do dicionário que mais se assemelha ao bloco da imagem de entrada em processamento.

O grau de semelhança entre um bloco da imagem e um bloco do dicionário é avaliado através do erro quadrático que corresponde à soma do quadrado das diferenças entre as intensidades dos pixéis do bloco de imagem em processamento e as intensidades dos pixéis correspondentes do bloco de dicionário em avaliação.

## Função encodeBlockimgX

## Função process\_block

## Função getNewTask

## Função build\_cod

## Função Write\_Z2\_cod\_to\_file

## Função validate\_dic\_pgm

## Função Write\_Z5\_cod\_to\_file

**Funções Principais – ficheiro main.c**

## Função main

A função *main* consiste na função principal que irá invocar as principais funcionalidades da aplicação, nomeadamente a conversão dos ficheiros de imagem, conforme enunciado a*presentados pelos professores.*

Esta função trata o *parsing* dos parâmetros passados pela linha de comandos.

Este processamento dos parâmetros do utilizador será realizado por intermédio da biblioteca

<*cmdline.h>* criada pelo ficheiro de configu*ração gengetopt pagegengetopt*.ggo.

Esta função é também responsável pela contabilização do tempo despendido em cada operação, recorrendo ao uso da biblioteca *<time.h>* e da função clock().

# - Conclusão

Este relatório tem como objetivo principal a documentação do código realizado e implementado demonstrando quais das funcionalidades do paCodec estão funcionais e a validação dos argumentos permitidos e/ou obrigatórios solicitados no enunciado.

Os exercícios realizados nas aulas foram fundamentais para a resolução desta segunda etapa do projeto, tornando cada vez mais aliciante o alcançar de todos os objetivos propostos.

Todos os objetivos foram atingidos para as várias funcionalidades exigidas no enunciado do projecto, no entanto entendemos que a solução atual apresentada, possa não ser a mais correta, uma vez que nesta área, várias opções possam ser tomadas para o mesmo fim, porque reiteramos mais uma vez que o projeto envolveu um elevado grau de complexidade e a nossa pouca experiência na linguagem C tornou o desenvolvimento desta etapa uma tarefa árdua de concretizar.

As principais dificuldades que encontrámos para a realização desta etapa consistiram na programação do algoritmo da codificação das imagens em formato .pgm, tanto para que algoritmo codificasse bem as imagens mas também que ao ser paralelizado em threads pudesse continuar a atingir os objectivos a que se propôs.

A elaboração desta etapa contribuiu para melhorar os nossos conhecimentos como estudantes e programadores, concluindo que a linguagem de programação C é bastante versátil e focalizada na programação procedimental.

# - Objetivos

## Objetivos Atingidos

* + Funcionalidade de codificar imagens em formato *.cod* do tipo Z2 a partir de ficheiros do tipo .pgm, implementada cm sucesso.
  + Funcionalidade de codificar imagens em formato *.cod* do tipo Z5 a partir de ficheiros do tipo .pgm, implementada cm sucesso.
  + Funcionalidade de codificar imagens em paralelo (usando threads) de ficheiros de formato *.cod* do tipo Z2 implementada cm sucesso.
  + Funcionalidade de codificar imagens em paralelo (usando threads) de ficheiros de formato *.cod* do tipo Z5 implementada cm sucesso.
  + Criação e gravação do ficheiro Z2 codificado *.cod* para a diretoria do ficheiro original *.pgm* , implementada com sucesso.
  + Criação e gravação do ficheiro Z5 codificado *.cod* para a diretoria do ficheiro original *.pgm* , implementada com sucesso.

## Objetivos Não Atingidos

* Nesta etapa todos os objectivos foram concluídos com sucesso

# - Bibliografia

As referências bibliográficas de seguida, foram de especial auxílio para a criação das caraterísticas do descodificador de imagens necessárias á 1ª etapa.

* + Apresentações das aulas teóricas/práticas de Programação Avançada
  + Exercícios práticos realizados nas Aulas práticas
  + Exercícios/Fichas práticas das aulas de Programação Avançada
  + Comando *man* (manual de instruções) usado nas funções passadas na linha de comandos de *Linux*
  + *Livro C por “Luís Damas*