

1º Semestre 16/17 SI

Comunicação e Processamento de Sinais

**Relatório do 5º Trabalho Prático**

Eng. Pedro Fazenda

28/1/2017

**Trabalho realizado por:**

Diogo Fernandes nº39205

Rui Santos nº39286

**Índice**

[**Introdução** 3](#_Toc473409562)

[**Amostragem** 3](#_Toc473409563)

[**Quantificador** 3](#_Toc473409564)

[**Codificador** 4](#_Toc473409565)

[**Controlo de Erros** 4](#_Toc473409566)

[**Modulação Digital** 4](#_Toc473409567)

[**Desenvolvimento** 5](#_Toc473409568)

[**QPSK** 5](#_Toc473409569)

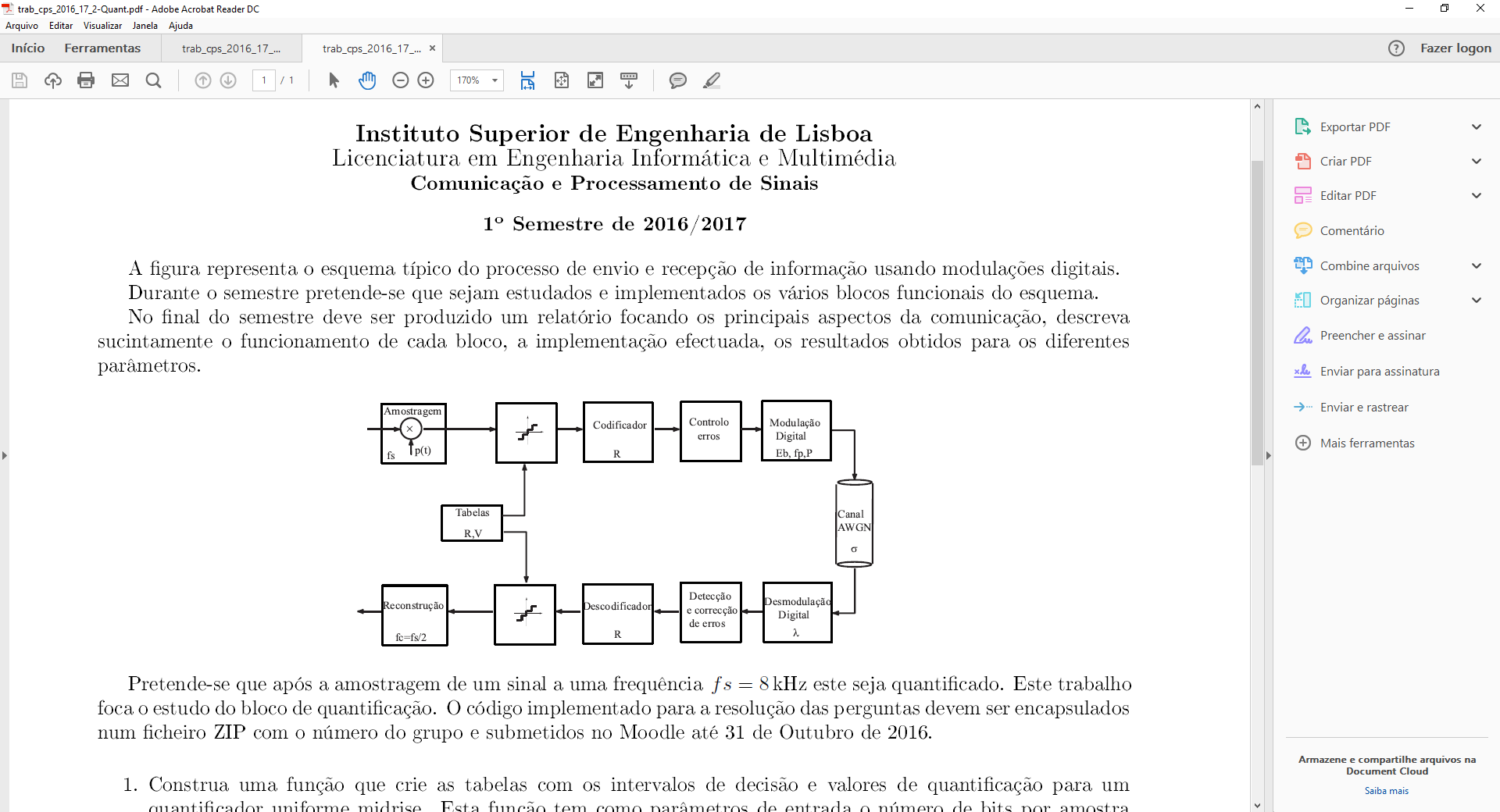
[**Receptor QPSK** 7](#_Toc473409570)

[**Sistema** 8](#_Toc473409571)

[**Avaliação do sistema** 10](#_Toc473409572)

[**Conclusão** 12](#_Toc473409573)

**Introdução**



Este quarto trabalho foca-se nos blocos de Modulação digital e suas operações inversas, para esse efeito iremos utilizar uma modulação M-Ária QPSK, *(Quadrant Phase Shifting Key)*.

Sendo esta a implementação final do nosso diagrama de blocos, iremos executar uma série de testes a todos os blocos em funcionamento conjunto.

## **Amostragem**

O Primeiro bloco consiste em amostrar o sinal, ou seja, torna-lo discreto no domínio do tempo. Esta função pode ser descrita como a observação do valor do sinal analógico de entrada a intervalos regulares. O sinal amostrado é obtido pelo produto entre o sinal de entrada e um trem de impulsos de *dirac* com período *Ts*. Dado que a amostragem corresponde à multiplicação no domínio do tempo, o espectro do sinal amostrado corresponde à convulação do espetro do sinal, que se supõe pela transformada de Fourier do trem de impulsos com período e área *fs = 1/Ts*.

## **Quantificador**

Quantificação de um sinal é o processo que converte um sinal amostrado (discreto no tempo), num sinal com valores também discretos em amplitude.

Primeiro obtém-se os valores discretos no tempo do sinal a quantificar, após isso, configura-se os intervalos de quantificação baseados no numero de Bits que queremos quantificar e o valor máximo dessa quantificação e verifica-se a que intervalo de quantificação corresponde o valor do sinal amostrado.

## **Codificador**

A Codificação nada mais é, do que a representação do sinal anteriormente quantificado e amostrado, numa sequência binária, com o mesmo numero de bits configurados na sua quantificação.

Após a quantificação, são guardados os índices dos níveis de quantificação utilizados no sinal quantificado, e convertidos de decimal para binário, obtendo assim o sinal codificado.

## **Controlo de Erros**

Para transmitir o nosso sinal através do canal AWGN são precisos bits para controlar a sua receção, de modo a haver menos erros quando os transmitimos no canal.  
Para o controlo de erros neste processo inteiro, iremos utilizar código Hamming Cíclico e não sistemático.

## **Modulação Digital**

No canal de comunicação, os sinais não são transmitidos instantaneamente, são atenuado devido à resistência elétrica, distorcidos devido à largura de banda e inseridos com ruido térmico ou interferências eletromagnéticas.

Como qualquer outra transmissão digital a BT (Largura de banda) é um recurso fundamental que interessa preservar a todo o custo. A eficiência espetral é a relação entre o débito binário e a largura de banda ocupada pelo sinal transmitido, servindo de medida de qualidade relação a estes atributos. A transmissão M-ária de símbolos com duas formas de ondas faz diminuir a transmissão de símbolos, mas mantendo o débito binário. A largura de banda é linearmente dependente do debito de símbolos, ora se este diminui, a largura de banda também, aumento a eficiência espetral na transmissão.

**Desenvolvimento**

## **QPSK**

1. Construa uma função que simule a modulação digital QPSK. Esta função recebe como parâmetro de entrada um array de bits e retorna um array com o sinal modulado a transmitir. Esta função ainda deve receber como parâmetros de entrada o número de pontos por cada bit (P=8) e a Energia média por bit (Eb). Admita que a frequência da portadora é um quarto do débito binario, ou seja, cada simbolo contém um período da portadora.

#dicionario das fases correspondentes aos bits transmitidos do codigo de HAMMING

fases **=** **{**"11"**:**np**.**pi**/**4**,**"01"**:**3**\***np**.**pi**/**4**,**"00"**:**5**\***np**.**pi**/**4**,**"10"**:**7**\***np**.**pi**/**4**}**

**def** QPSK**(**arrayBits**,**P**,**Eb**):**

#sabemos que é uma modelação com 4 simbolos

#logo assumimos que a cada simbolo agrupamos 2 bits

nBits **=** 2

#iterador do array de fases

currentPosition **=** 0

#calculo da amplitude constante

A **=** np**.**sqrt**(**Eb**\***4**)**

#fases correspondentes aos bits

arrayFases **=** np**.**zeros**(**int**(**len**(**arrayBits**)/**2**))**

**for** i **in** range**(**0**,**len**(**arrayBits**),**nBits**):**

currentBits **=** arrayBits**[**i**:**i**+**nBits**]**

**if(**currentBits**[**0**]==**0**):**

**if(**currentBits**[**1**]==**0**):**

arrayFases**[**currentPosition**]** **=** fases**[**"00"**]**

**else:**

arrayFases**[**currentPosition**]** **=** fases**[**"01"**]**

**else:**

**if(**currentBits**[**1**]==**0**):**

arrayFases**[**currentPosition**]** **=** fases**[**"10"**]**

**else:**

arrayFases**[**currentPosition**]** **=** fases**[**"11"**]**

currentPosition **+=**1**;**

secondArray **=** np**.**zeros**(**len**(**arrayFases**)\***P**)**

aux **=** 0

**for** i **in** range**(**len**(**arrayFases**)):**

**for** j **in** range**(**0**,**P**):**

calculo **=** A**\***np**.**cos**(((**2**\***np**.**pi**\***j**)/**P**)** **+** arrayFases**[**i**])**

secondArray**[**aux**]** **=** calculo

aux **+=** 1

**return** secondArray

## **Receptor QPSK**

**2.** Construa uma função que simule, no receptor, um filtro adaptado (desmodulação). Este deve receber um array com o sinal QPSK com ruído e deve retornar um array com a sequência binária correspondente.

**def** desmodulacao**(**arrayQPSK**,**P**):**

#ritmo de simbolos

Rs **=** 1**/**P

valueX **=** 0

valueY **=** 0

finalArrayX **=** np**.**zeros**(**P**)**

finalArrayY **=** np**.**zeros**(**P**)**

arrayQPSKdesmodulado **=** np**.**zeros**(**int**((**len**(**arrayQPSK**)/**P**)\***2**))**

**for** i **in** range**(**int**(**len**(**arrayQPSKdesmodulado**)/**2**)):**

current **=** arrayQPSK**[**i**\***P**:(**i**+**1**)\***P**]**

**for** j **in** range**(**0**,**P**):**

calculoY **=** **-**np**.**sqrt**(**2**\***Rs**)\***np**.**sin**((**2**\***np**.**pi**\***j**)/**P**)\***current**[**j**]**

calculoX **=** np**.**sqrt**(**2**\***Rs**)\***np**.**cos**((**2**\***np**.**pi**\***j**)/**P**)\***current**[**j**]**

plt**.**plot**(**calculoX**,**calculoY**,**'o'**)**

finalArrayY**[**j**]** **=** calculoY

finalArrayX**[**j**]** **=** calculoX

#somatorio do array

valueY **=** sum**(**finalArrayY**)**

valueX **=** sum**(**finalArrayX**)**

#construção do array final

**if(**valueX **>** 0**):**

arrayQPSKdesmodulado**[**i**\***2**]** **=** 1

**else:**

arrayQPSKdesmodulado**[**i**\***2**]** **=** 0

**if(**valueY **>** 0**):**

arrayQPSKdesmodulado**[(**i**\***2**)+**1**]** **=** 1

**else:**

arrayQPSKdesmodulado**[(**i**\***2**)+**1**]** **=** 0

##reset aos arrays para não iterar

finalArrayX **=** np**.**zeros**(**P**)**

finalArrayY **=** np**.**zeros**(**P**)**

plt**.**grid**(True)**

plt**.**ylim**([-**1.5**,**1.5**])**

plt**.**xlim**([-**1.5**,**1.5**])**

plt**.**show**()**

**return** arrayQPSKdesmodulado**.**astype**(**"int16"**)**

## **Sistema**

**3.** Simule o sistema de transmissão digital com base nas funções contruídas durante o semestre. Este sistema tem as seguintes características:

• Sinal de entrada é um sinal de áudio (móno) adquirido a uma frequência de amostragem de 8kHz com 16 bits por amostra;

• Codificador PCM, com quantificador uniforme midrise, parâmetros: Número de bits de codificação R = 8 e frequência de amostrasgem fs = 8000;

• Controlo de erros: Código de Hamming H(7,4). Nota: o sistema deve estar preparado para funcionar com e sem este módulo.

• Modulação digital QPSK, com parâmetros: Número de amostras para cada símbolo P = 8 e energia média por bit Eb.

• Canal AWGN, parâmetro: Potência do ruído ;

Para que pudéssemos simular este sistema, criamos uma função onde todas as funções dos módulos do sistema fossem chamadas para cumprir a sua parte dando um resultado final de ouvir-se o sinal em diferentes potencias de ruido aplicadas no canal.

**def** sistema**():**

P **=** 8

Eb **=** 1

Pr **=** **[**0.5**,**1**,**2**,**4**]**

berTeoricoBefore **=** np**.**zeros**(**len**(**Pr**))**

berPraticoBefore **=** np**.**zeros**(**len**(**Pr**))**

snrPratica **=** np**.**zeros**(**len**(**Pr**))**

berTeoricoAfter **=** np**.**zeros**(**len**(**Pr**))**

berPraticoAfter **=** np**.**zeros**(**len**(**Pr**))**

fsRecord**,** data **=** wav**.**read**(**caminho**+**"sinaldevoz8khz.wav"**)**

R **=** 8

**for** i **in** range**(**len**(**Pr**)):**

#criacao das tabelas

VD**,**NQ **=** createTable**(**R**,**np**.**max**(**np**.**abs**(**data**)))**

#quantificacao do sinal lido

SQ**,** IQ **=** quantificacao**(**data**,**NQ**,**VD**)**

#codificacao do sinal quantificado

signalCodif **=** codificaSinal**(**IQ**,**R**)**

#adicao dos bits de erro

signalControl **=** hamming**(**signalCodif**)**

#gerar o QPSK

sinalQPSK **=** QPSK**(**signalControl**,**P**,**Eb**)**

#passagem pelo canal

canal **=** canalAWGN**(**sinalQPSK**,**Pr**[**i**])**

#aplicação da desmodulacao

desmodulado **=** desmodulacao**(**canal**,**P**)**

#correcao hamming

correctedB**,** corrected **=** sindrome**(**desmodulado**)**

#descodificacao do sinal

decodedSignal **=** descodificaSinal**(**corrected**,**R**)**

#desquantificacao do sinal

signalQuant **=** quantificacaoInversa**(**decodedSignal**,**NQ**)**

################ ---- ERROS ------ ##########################

####----------medicao de BER's------------####

berTeoricoBefore**[**i**]** **=** BERteorico**(**signalCodif**,**corrected**)**

berPraticoBefore**[**i**]** **=** BERpratico**(**signalCodif**,**corrected**)**

berTeoricoAfter**[**i**]** **=** 0.5**\***erfc**(**np**.**sqrt**(**Eb**/**Pr**[**i**]))**

berPraticoAfter**[**i**]** **=** np**.**sum**(**1**\***np**.**logical\_xor**(**signalControl**,**desmodulado**))/**len**(**desmodulado**)**

## medicao da SNRPratica

erro **=** erroQuantificacao**(**data**,**signalQuant**)**

potenciaErro **=** potenciaErroQuant**(**erro**)**

potencia **=** potenciaSinal**(**data**)**

snrPratica**[**i**]** **=** SNRPratico**(**potencia**,**potenciaErro**)**

recordSignal**(**"sinal-ruido-"**+**str**(**Pr**[**i**])+**".wav"**,**fsRecord**,**signalQuant**.**astype**(**'int16'**))**

## **Avaliação do sistema**

**4.**Na avaliação do sistema deve medir experimentalmente e comparar, sempre que possível, com os valores teóricos, os seguintes índices:

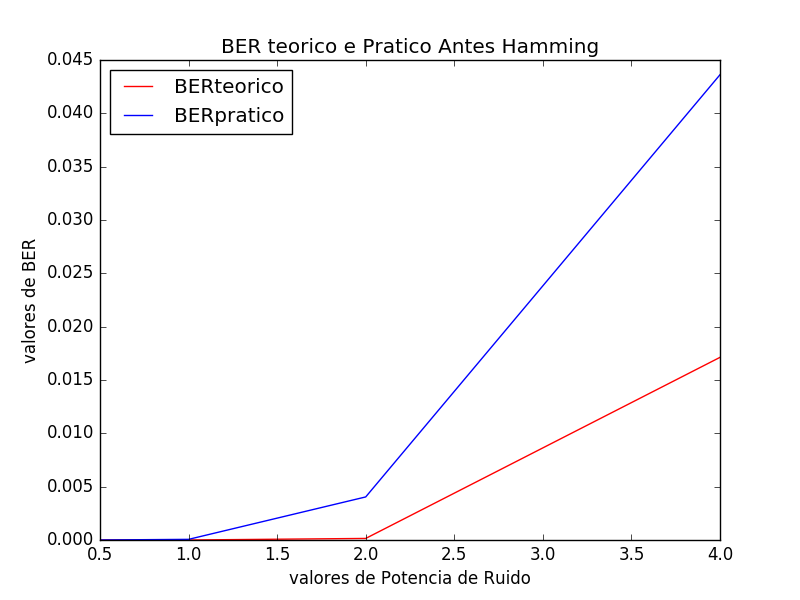
• A SNR no canal;

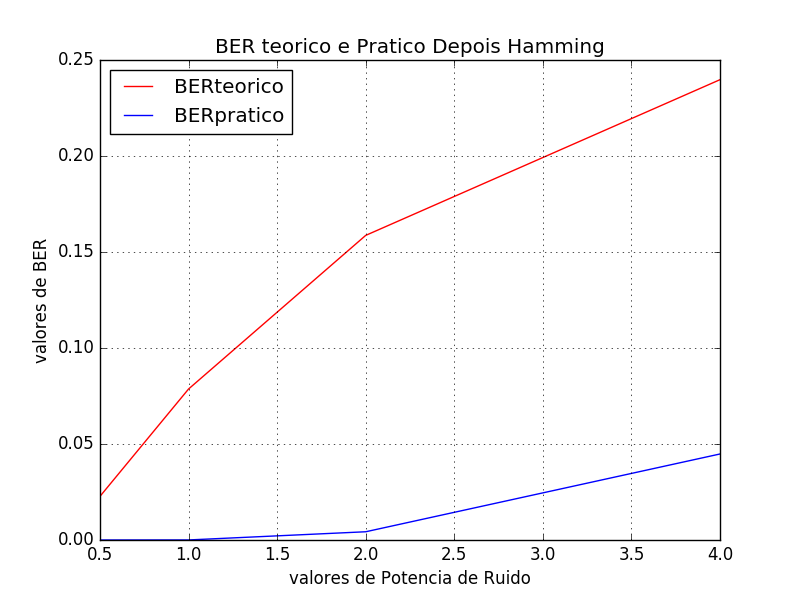
• A SNR na recepção ( neste caso considera-se que o ruído é a diferença entre o sinal à entrada do transmissor e o sinal à saída do receptor);

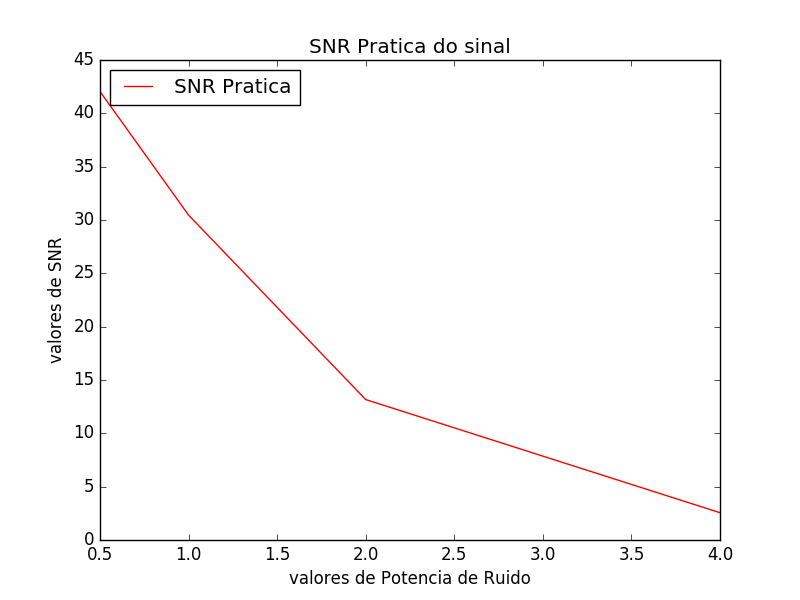
• BER antes da correcção de erros;

• BER depois da correcção de erros;

• Apresente a constelação antes e depois do sinal passar pelo canal. Para o efeito deve atribuir diferentes valores aos parâmetros do sistema e avalia-lo com e sem o módulo de correcção de erros.







# **Conclusão**

Na realização deste projeto durante o ano, existiram diversas dificuldades na compreensão de alguns conceitos de acordo com os sinais.

No desenvolvimento deste bloco, Modulação digital, conseguimos perceber da mesma maneira que foi desenvolvida no trabalho do Código Linha, que a potência de ruído tem um efeito bastante grande na transmissão dos valores modulados a QPSK.  
Nos diferentes sinais gravados, conseguimos perceber que apenas quando a potência de ruído atinge o valor 2, denotamos um certo ruído, mas ainda compreensível o conteúdo do ficheiro de áudio, ao simularmos com a potência de valor 4, o ruído já é bastante acrescido sendo ainda possível a compreensão do conteúdo do ficheiro de áudio, mas com baixa qualidade.

Os valores de SNR parecem um bocado fora do normal, não percebendo a origem de tais erros, pois à priori haviam sido testadas as funções e apresentaram valores corretos.

Não foram atingidos dois objetivos do trabalho por dificuldades na compreensão:

- Constelação correspondente antes do canal e depois do canal.

- SNR do canal

Mesmo não tendo cumpridos os requisitos anteriormente referidos, notamos que o sistema corre sem falhas e obtemos os resultados esperados.