deti universidade de aveiro
departamento de eletrónica, telecomunicações e informática

Taxas de Leitura/Escrita de processos em *Bash*

Professor: Nuno Lau (nunolau@ua.pt)

Marta Inácio Nº101726 (50%)
Gabriel Teixeira Nº107876 (50%)

Sistemas Operativos **Ano letivo 2022/2023**

Introdução ao trabalho

No contexto da cadeira de Sistemas Operativos, foi-nos atribuída a realização deste trabalho prático, que consiste na visualização e tratamento dos processos que estão a decorrer no aparelho utilizado. O objetivo deste trabalho é o desenvolvimento de um script em bash para obter estatísticas sobre as leituras e escritas que os processos estão a efetuar. Esta ferramenta permite visualizar o número total de bytes de I/O que um processo leu/escreveu e também a taxa de leitura/escrita correspondente aos últimos s segundos para uma seleção de processos.

A implementação do código foi realizada através do software Gedit e do editor de código ,IDE VSCode, pois são ambos os membros do grupo estão familiarizados com o uso dos mesmos.

O script rwstat.sh será feito utilizando matéria dada nas aulas práticas, onde foram abordados os conceitos base de programação em *bash*, que nos permitem listar os processos que se encontram a correr no nosso computador.

Tal como discriminado no guião, temos de criar vários comandos que filtrem a utilização da script, desde a utilização de opções para a ordenação dos processos e da seleção de um conjunto específico de processos. A nossa ideia era guardar a informação utilizando arrays associativos, visto que são estruturas de dados onde é possível identificar cada elemento por uma key, e por um value. Depois de ter a informação dos processos toda tratada/formatada e devidamente guardada, iremos criar as expressões condicionais para tratar a informação de acordo com as opções passadas no terminal, quando se corre o programa.

.

Apresentação do código

Declaração de variáveis globais

Tal como mencionado anteriormente, começámos por declarar arrays associativos para guardar a informação relativa aos processos. Para além disso, criámos diversas variáveis globais utilizadas que terão diversas utilizações. Assim inicializamos as variáveis **segundos, num, nProc, r** e os arrays ,**procs, args, read** e **write**.

```
segundos=${@: -1}  #Variavél que contém o argumento passado na linha de comandos correspondente ao número de segundos
num='^[0-9]?$'     #Variavél que contém a expressão regex para verificar se o argumento passado é um número inteiro
nProc=0            #Variavél correspondente ao número de processos a mostrar no terminal
r=0                #Variavél utilizada para definir ordem de ordenação da Tabela no terminal

declare -A procs=()   #Array associativo onde está guardada a informação de cada processo, para aceder á informação de cada processo usamos o PID
declare -A args=()    #Array associativo onde está guardada a informação das opções passadas como argumentos
declare -A read=()    #Array associativo onde está guardada a informação do valor de leitura de bytes de cada processo
declare -A write=()   #Array associativo onde está guardada a informação do valor de escrita de bytes de cada processo
```

(Fig 1)

- **segundos** - Variável que contém o argumento passado na linha de comandos correspondente ao número de segundos.

- **num** - Variável que contém a expressão regex para verificar se o argumento passado é um número inteiro.

- **nProc** - Variável correspondente ao número de processos a mostrar no terminal.

- **r** - Variável utilizada para definir a ordem de ordenação da Tabela no terminal.

- **procs** - array associativo que contém informações acerca de todos os processos que,posteriormente tenham passado pelos processos de validação no nosso código, tem como key o PID de cada processo, não havendo, assim, colisão de informação.

- **args** - array associativo onde estão guardadas as informações das opções passadas como argumento, quando se corre o programa, sendo a key, a opção passada pelo utilizador e o seu valor o argumento da mesma.

- **read** - array associativo onde está guardada a informação sobre o número total de bytes de I/O a ler.

- **write** - array associativo onde está guardada a informação sobre o número total de bytes de I/O a escrever.

Tratamento de opções

Para o tratamento de opções criámos a função menu, para que o utilizador possa ver todas as funcionalidades possíveis , utilizamos o comando *getopts* e alguns testes condicionais para verificar erros de utilização. As diversas operações válidas apresentadas na função menu, são: *-c ,-u ,-s ,-e,-m,-M,-w,-p,-r .*

```
20 function menu() {       #Função que lista todas as opções de utilização válidas
21     echo ""
22     echo "                                    Menu
23     echo "Opções de seleção de processos:"
24     echo ""
25     echo "     -c      -> Seleção por uma expressão regular"
26     echo "     -u      -> Seleção pelo nome do utilizador"
27     echo "     -s      -> Seleção por um periodo temporal(data mínima)"
28     echo "     -e      -> Seleção por um periodo temporal(data máxima)"
29     echo "     -m      -> Seleção por gama de PID(Pid minimo)"
30     echo "     -M      -> Seleção de processos da tabela por gama de PID- Pid máximo"
31     echo "     -p      -> Seleção de número de processos a visualizar"
32     echo ""
33     echo "Opção de ordenação da tabela:"
34     echo ""
35     echo "     -w      -> Ordenação da tabela por valores do write(ratew)"
36     echo "     -r      -> Ordenação reversa"
37 }
```

(Fig 2)

- **(-c) -** Esta opção permite filtrar os processos através de uma expressão *regex* que o utilizador passa como os argumentos de operação, mostrando apenas os processos que o utilizador que verificam essa condição.
- **(-u) -** Esta opção permite a seleção de processos através do nome do utilizador.
- **(-s)-** Esta opção permite a seleção de processos pela especificação de um período temporal, neste caso através da data mínima, apresentando apenas os processos que ocorreram posteriormente a essa data.
- **(-e) -** Esta opção permite a especificação de um período temporal, neste caso através da data máxima, apresentando apenas os processos que ocorreram anteriormente a essa data.
- **(-m)-** Esta opção permite a seleção de processos através de uma gama de PID , sendo neste caso limitada inferiormente.
- **(-M) -** Esta opção permite a seleção de processos através de uma gama de PID , sendo neste caso limitada inferiormente.
- **(-p) -** Esta opção permite obter o número específico de processos a visualizar escolhido pelo utilizador.
- **(-w)-** Esta opção permite ordenar a tabela apresentada no terminal pelo valor do RateW dos processos.
- **(-r)-** Esta opção permite ordenar a tabela apresentada no terminal pela ordem inversa.

Getopts

De seguida temos o *getopts,* figura abaixo que nos vai permitir passar várias opções e fazer o seu tratamento.

```
39
40 while getopts "c:u:s:e:m:M:p:wr" option; do   #Análise da informação passada por cada argumento de entrada
41
```

(Fig 3)

Dentro do comando *getopts* iremos prosseguir à adição das opções passadas e dos seus respetivos argumentos. Caso a opção passada não possua nenhum argumento, a key que vai ficar no array args vai ser essa mesma opção, e caso não tenha nenhum argumento então vai ficar com o seu value igual a “nada”. Por outro lado, se a opção passada tivesse argumento, a key que vai ficar no array args vai ser essa mesma opção, contudo, o seu value, desta vez, vai ser o argumento passado com a opção.

```
#Guarda as opções passadas ao correr a script, guardando "nada" nas opções que não são passadas
if [[ -z "$OPTARG" ]]; then
    args[$option]="nada"
else
    args[$option]=${OPTARG}
fi
```

(Fig 4)

Como podemos observar na figura abaixo , efetuamos o tratamento de cada opção possível. Para opções como “-r” e “-w”, apenas guardamos o valor do argumento para futura utilização.
Para as restantes operações verificamos se a opção tem um argumento válido, em muitos casos se é um número inteiro, noutros se é uma data válida, ou até se é uma string e tem todos os argumentos válidos

Supondo que o programa foi chamado com as opções -m -r -i 10, o getopts ao receber a opção “-m” vai ver se o valor de i é igual a um e caso não seja vai meter i igual a um, na segunda opção vai iterar com a opção “–r” que é uma opção válida para usar com “-m”, o getopts não vai fazer nada na opção “-r” visto que esta não tem nada para fazer, em seguida vai à opção “-i” e como esta não se verifica com os outros casos então o programa vai chamar a função menu() e vai terminar o programa.

```
#Tratamentos das opçoes passadas como argumentos
while getopts "c:u:s:e:m:M:wp:r" option; do

    #Adicionar ao array argOpt as opcoes passadas ao correr o procstat.sh, caso existam adiciona as que são passadas, caso não, adiciona "nada"
    if [[ -z "$OPTARG" ]]; then
        argOpt[$option]="nada"
    else
        argOpt[$option]=${OPTARG}
    fi

    case $option in
    c) #Seleção de processos a utilizar atraves de uma expressão regular
        str=${argOpt['c']}
        if [[ $str == 'nada' || ${str:0:1} == "-" || $str =~ $re ]]; then
            echo "Argumento de '-c' não foi preenchido, foi introduzido argumento inválido ou chamou sem '-' atrás da opção passada." >&2
            opcoes
            exit 1
        fi
        ;;
    u) #Seleção de processos a visualizar através do nome do utilizador
        str=${argOpt['u']}
        if [[ $str == 'nada' || ${str:0:1} == "-" || $str =~ $re ]]; then
            echo "Argumento de '-u' não foi preenchido, foi introduzido argumento inválido ou chamou sem '-' atrás da opção passada." >&2
            opcoes
            opcoes exit 1
        fi
        ;;
    s) #Seleção de processos a visualizar num periodo temporal - data mínima
        str=${argOpt['s']}
        regData='^((Jan(uary)?|Feb(ruary)?|Mar(ch)?|Apr(il)?|May|Jun(e)?|Jul(y)?|Aug(ust)?|Sep(tember)?|Oct(ober)?|Nov(ember)?|Dec(ember)?)) +[0-9]{1,2} +[0-9]
        if [[ $str == 'nada' || ${str:0:1} == "-" || $str =~ $re || ! "$str" =~ $regData ]]; then
            echo "Argumento de '-s' não foi preenchido, foi introduzido argumento inválido ou chamou sem '-' atrás da opção passada." >&2
            opcoes
            exit 1
        fi
        ;;
    e) #Seleção de processos a visualizar num periodo temporal - data máxima
        str=${argOpt['e']}
        regData='^((Jan(uary)?|Feb(ruary)?|Mar(ch)?|Apr(il)?|May|Jun(e)?|Jul(y)?|Aug(ust)?|Sep(tember)?|Oct(ober)?|Nov(ember)?|Dec(ember)?)) +[0-9]{1,2} +[0-9]
        if [[ $str == 'nada' || ${str:0:1} == "-" || $str =~ $re || ! "$str" =~ $regData ]]; then
            echo "Argumento de '-e' não foi preenchido, foi introduzido argumento inválido ou chamou sem '-' atrás da opção passada." >&2
            opcoes
            exit 1
        fi
        ;;
    m) #seleção da tabela por gama de PID- Pid minimo
        str=${argOpt['m']}
        if ! [[ ${argOpt['m']} =~ $re ]]; then
            echo "Argumento de '-p' tem de ser um número ou chamou sem '-' atrás da opção passada." >&2
            opcoes
        fi
        ;;

    #Verificação que o argumento passado com a opção -M é válido para a seleção de processos por uma gama de PID, neste caso PID máximo
    M)  pidMax=${args['M']}
        if ! [[ $pidMax =~ $num ]]; then
            echo "Argumento de '-M' tem de ser um número ou chamou sem '-' atrás da opção passada." >&2
            menu
            exit 1
        fi
        ;;

    #Verificação que o argumento passado com a opção -p é válido para a seleção do número de processos a apresentar no terminal
    p)  nProc=${args['p']}
        if ! [[ ${nProc} =~ $num ]]; then
            echo "Argumento de '-p' tem de ser um número inteiro superior a 0" >&2
            menu
            exit 1
        fi
        ;;

    w) #Não há verificação, pois a opção não aceita argumento

        ;;

    #Argumento para a ordenação da tabala a imprimir no terminal por ordem inversa
    r)  r=1
        ;;

    #Apresenta o menu e termina com a passagem de argumentos inválidos
    *)  menu
        exit 1
        ;;
    esac

done
```

Como podemos observar na seguinte imagem, fizemos uma verificação que nos permitiu ver se o último argumento passado correspondente aos segundos é um número e ao mesmo tempo se é diferente de zero, pois não ocorrem processos num intervalo de tempo nulo.

```
# Verifica se o último argumento passado é um número e é diferente de zero
if ! [[ $segundos =~ $num && $segundos != 0 ]]; then
    echo "Último argumento correspondente aos segundos tem de ser um número inteiro positivo">&2
    menu
    exit 1
fi
```

(Fig 4)

Seguidamente iniciamos um ciclo for, este ciclo for, será somente para realizarmos uma leitura das variáveis rchar e wchar, que cada processo dispõe desta maneira conseguimos fazer sleep uma vez em toda a execução do script. No ciclo for, vamos percorrer todos os processos ativos no aparelho, em que o nome da diretoria são somente números .

De seguida aplicamos uma condição para verificar se temos permissões no status, e no io, no determinado processo em que o ciclo for itera, tendo permissões, guardamos na variável PID, o pid do processo, utilizamos um grep, e o tr de modo a que só retornasse os números
correspondentes ao pid.

Guardamos nas variáveis rchar1 e wchar1 os valores correspondentes ao rchar e ao
wchar, respetivamente, utilizamos um grep e um tr de modo a só retornar para a função os valores correspondentes ao que queremos.

Prosseguindo no código, encontramos um if, este if é essencial caso os valores de casoos valores de rchar1 e wchar1 sejam 0, se porventura isso acontecer utilizamos a palavra reservada continue, que faz avançar para o próximo processo no ciclo for. Caso as condições não se verifiquem, adicionamos aos arrays associativos read, e write, os valores de rchar1 e wchar1, respetivamente, onde a key pela qual guardamos os valores nos arrays é o pid, calculado logo no início deste bloco de código.

```
for entry in /proc/[[:digit:]]*; do
    if [[ -r $entry/status && -r $entry/io ]]; then
        PID=$(cat $entry/status | grep -w Pid | tr -dc '0-9') # Obter o PID
        rchar1=$(cat $entry/io | grep rchar | tr -dc '0-9')   # Obter o rchar inicial
        wchar1=$(cat $entry/io | grep wchar | tr -dc '0-9')   # Obter o wchar inicial

        if [[ $rchar1 == 0 && $wchar == 0 ]]; then
            continue
        else
            read[$PID]=$(printf "%12d\n" "$rchar1")
            write[$PID]=$(printf "%12d\n" "$wchar1")
        fi
    fi

done
```

(Fig 5)

Ao fim de todos os processos serem lidos pelo for, este termina e executamos o comando
sleep, onde lhe passamos o argumento $segundos, definido no início da script.

```
sleep $segundos # tempo de espera até á leitura seguinte
```

(Fig 6)

Executamos agora outro ciclo for sendo que é neste em que a maioria da informação vai ser extraída, este ciclo itera sobre os processos que estão ativos na nossa máquina, ou seja, itera da mesma maneira que o for anterior. De seguida aplicamos uma condição para verificar se temos permissões no status, e no I/O, no determinado processo em que o ciclo for itera.

Vamos extrair o pid, para a variável PID, através de um grep e de um tr, de modo a que só fique guardado na variável o número do mesmo.

```
for entry in /proc/[[:digit:]]*; do
    if [[ -r $entry/status && -r $entry/io ]]; then

        PID=$(cat $entry/status | grep -w Pid | tr -dc '0-9') # Obter o PID
```

Ainda dentro deste ciclo for vamos testar algumas condições que nos vão permitir apresentar apenas determinados valores no terminal, tal como escolhido pela opção introduzido ao executar a script rwstat.sh. Como por exemplo, a seleção por um gama de PID.

Como se verifica na seguinte imagem, para filtrar os processos por uma gama de PID, temos um if, que verifica se as opções adequadas se encontram no array associativo args. Inicialmente verificamos se ambas as opções “-m” e “-M” estão a ser utilizadas adequadamente, se o pid mínimo é inferior ao pid máximo. Por fim, realizamos dois ifs para averiguar se o processo atual tem um pid mínimo e/ou máximo é superior e/ou inferior ao passado como argumento nas opções. Caso não seja, descartamos o processo através do comando continue.

```
#seleção por gama de PID maximo e minimo
if [[ -v args[m] && -v args[M] ]]; then
        if [[ "${pidMin}" -gt "${pidMax}" ]]; then
            echo "O PID minimo é superior ao máximo, gama inválida">&2
        menu
        exit 1
        fi
    fi
```

```
#seleção por gama de PID minimo
if [[ -v args[m] ]]; then
        if [[ "$PID" -lt "${pidMin}" ]]; then
            continue
        fi
    fi

    #seleção por gama de PID maximo
    if [[ -v args[M] ]]; then
        if [[ "$PID" -gt "${pidMax}" ]]; then
            continue
        fi
    fi

user=$(ps -o user= -p $PID)         #Obter o utilizador correspomdente ao PID

    comm=$(cat $entry/comm | tr " " "_") # Obter o comm,e retirar os espaços e substituir por '_' nos comm's com 2 nomes
```

Seguidamente e através de expressões condicionais fomos tentar otimizar o nosso código, ou seja, se ao correr o script, foram passadas as opções “-c” ou “-u”, temos de filtrar os processos correspondentes as essas opções e aos seus respetivos argumentos, de modo a que só sejam adicionados processos que o utilizador ao correr o script pediu.

Ora, para a opção “-u”, temos um if, que verifica se essa opção se encontra no array
associativo args, que como já foi explicado anteriormente é um array onde ficam guardadas as opções e os argumentos das mesmas passadas ao correr o script, e se o user do processo em questão for diferente daquele que foi passado à opção

“-u”, então passamos para o próximo processo, descartando assim o anterior através do comando continue.

Para a opção “-c”, o procedimento e o raciocínio é o mesmo utilizado na opção anterior,vamos ver se o args tem o “-c” como key, e se o comm do processo for diferente daquele que foi passado no terminal quando se correu o script, é usado o comando continue, descartamos este processo, e passamos para o próximo.

```
#seleção pelo utilizador
    if [[ -v args[u] && ! ${util} == $user ]]; then
        continue
    fi

    #seleção de processos a utilizar atraves de uma expressão regular
    if [[ -v args[c] && ! $comm =~ ${padrao} ]]; then
        continue
    fi

    LANG=en_us_8859_1
    startDate=$(ps -o lstart= -p $PID)                                          # data de início do processo atraves do PID
    startDate=$(date +"%h %d %H:%M" -d "$startDate")
    dateSeg=$(date -d "$startDate" +"%h %d %H:%M"+%s | awk -F '[+]' '{print $2}') # data do processo em segundos

    #Seleção por data máxima e minima
    if [[ -v args[s] && -v args[e] ]]; then
        start=$(date -d "${dateMin}" +"%h %d %H:%M"+%s | awk -F '[+]' '{print $2}') # data mínima
        end=$(date -d "${dateMax}" +"%h %d %H:%M"+%s | awk -F '[+]' '{print $2}')

        if [[ "$start" -gt "$end" ]]; then
            echo "As datas introduzidas são incompatíveis, a data final tem de ser uma data posterior á data inicial">&2
        menu
        exit 1
        fi
    fi
```

Na parte das datas, datas dos processos, e data passadas como argumentos. Primeiramente, passamos o código para que a data fique no formato inglês, de seguida, através do comando ps e do PID, já calculado anteriormente, guardamos na variável startDate, a data de início do processo, a seguir passamos a data para o
formato “mmm dd HH:MM”, ou seja, mês no diminutivo, dia, hora, e minutos, pois é neste
formato que a data é apresentada no guião do trabalho prático. Na linha seguinte de código,
passamos para segundos a data de início do processo, pois será útil, para comparações futuras.

Para a filtragem por data mínima e/ou máxima, procedemos de forma semelhante á filtragem dos processos por gama de pid.

```
#Seleção por data minima
if [[ -v args[s] ]]; then
    start=$(date -d "${dateMin}" +"%h %d %H:%M"+%s | awk -F '[+]' '{print $2}')

    if [[ "$dateSeg" -lt "$start" ]]; then
        continue
    fi
fi

 #Seleção por data máxima
if [[ -v args[e] ]]; then
    end=$(date -d "${dateMax}" +"%h %d %H:%M"+%s | awk -F '[+]' '{print $2}')

    if [[ "$dateSeg" -gt "$end" ]]; then
        continue
    fi
fi
```

Seguidamente, fomos verificar quais os valores do rchar e do wchar após o tempo de espera criado pelo sleep e guardamos os valores nas variáveis rchar2 e wchar2. Depois subtraímos o valor inicial e o final do rchar e dividimos pelo número de segundos, para obter assim o RateR, RateW, ReadB e o WriteB.

```
rchar2=$(cat $entry/io | grep rchar | tr -dc '0-9') #valor do rchar após o tempo de espera
wchar2=$(cat $entry/io | grep wchar | tr -dc '0-9') #valor do wchar após o tempo de espera
subr=$(($rchar2-${read[$PID]})) #valor do ReadB, que é a diferença entre a primeira e segunda leitura do valor de rchar
subw=$(($wchar2-${write[$PID]})) # valor do WriteB, que é a diferença entre a primeira e segunda leitura do valor de wchar
rateR=$(echo "scale=2; $subr/$segundos" | bc -l) # calculo do rateR
rateR=${rateR/#./0.}
rateW=$(echo "scale=2; $subw/$segundos" | bc -l) # calculo do rateW
ratew=${ratew/#./0.}
```

Após a obtenção dos valores referidos anteriormente, possuímos todos os valores necessários para guardar no array associativo procs.

```
procs[$PID]=$(printf "%-27s %-16s %15d %12d %12d %15s %15s %16s\n" "$comm" "$user" "$PID" "$subr" "$subw" "$rateR" "$rateW" "$startDate")
```

Tendo todos os valores adequados relativos aos processos, tal como pedido pelo enunciado, podemos adiantar-nos á impressão do cabeçalho e dos valores na tabela.

```
#Cabeçalho
printf "%-27s %-16s %15s %12s %12s %15s %15s %16s\n" "COMM" "USER" "PID" "READB" "WRITEB" "RATER" "RATEW" "DATE"
```

Para a impressão da tabela temos de verificar se existia alguma opção de ordenação selecionada e, caso exista, ordenar a tabela dessa forma. Criámos um if para a opção “-p” caso ela esteja selecionada, o número de processos é o definido pelo argumento passado com a opção. Se não estiver selecionada, serão impressos todos os processos presentes no array procs.

```
#Seleção do número de processos a imprimir no cabeçalho
if [[ ${nProc} -eq 0 ]]; then
        nProc=${#procs[@]}
    fi
```

A ordenação da tabela em si é permitida pela variável r definida no início e pela opção “-w”, caso a opção “-r” esteja selecionada a tabela é ordenada de forma inversa, pelo nome ou pelo RateW caso a opção “-w” também esteja selecionada. Na hipótese de apenas “-w” ter sido usada a ordenação vai ser efetuada de forma crescente de RateW. Para isso, usamos o -rn para a ordenação inversa e o -k1 ou -k7, consoante a coluna pela qual queremos a ordenação.

```
#Ordenação da tabela
if [[ -v args[w] && $r -eq 0 ]]; then
    printf '%s \n' "${procs[@]}" | sort -n -k7 | head -n ${nProc}
elif [[ -v args[w] && $r -eq 1 ]]; then
    printf '%s \n' "${procs[@]}" | sort  -rn -k7 | head -n ${nProc}
elif [[ $r -eq 1 ]]; then
    printf '%s \n' "${procs[@]}" | sort  -rn -k1 | head -n ${nProc}
else
    printf '%s \n' "${procs[@]}" | sort -k1 | head -n ${nProc}
fi
```

Alguns resultados

O primeiro resultado, é também o mais fácil, é executar o script, e passar só como argumento o número de segundos em que o sleep irá estar ativo.

```
gabriel@gabriel-HP-ENVY-Laptop-14-eb0xxx:~/Desktop/SO/Projeto1$ ./rwstat.sh 10
COMM                 USER                 PID        READB        WRITEB        RATER        RATEW             DATE
Isolated_Web_Co      gabriel             3170          223          223        22.30        22.30     Dec 02 18:33
Isolated_Web_Co      gabriel            13224           38           38         3.80         3.80     Dec 02 20:08
Isolated_Web_Co      gabriel            13378            6            6         0.60          .60     Dec 02 20:09
Isolated_Web_Co      gabriel            13489           16           16         1.60         1.60     Dec 02 20:10
Isolated_Web_Co      gabriel            13812            6            6         0.60          .60     Dec 02 20:16
Isolated_Web_Co      gabriel            18907          129          129        12.90        12.90     Dec 02 20:23
Isolated_Web_Co      gabriel            19832           63           63         6.30         6.30     Dec 02 20:40
Privileged_Cont      gabriel             2967            6            6         0.60          .60     Dec 02 18:33
RDD_Process          gabriel             3228            2            2         0.20          .20     Dec 02 18:33
Socket_Process       gabriel             2923            2            2         0.20          .20     Dec 02 18:33
Utility_Process      gabriel             3230            4            4         0.40          .40     Dec 02 18:33
WebExtensions        gabriel             3016            6            6         0.60          .60     Dec 02 18:33
Web_Content          gabriel            29924            8            8         0.80          .80     Dec 02 22:10
Web_Content          gabriel            29965            4            4         0.40          .40     Dec 02 22:11
Web_Content          gabriel            29993            2            2         0.20          .20     Dec 02 22:11
Xorg                 gabriel             1562         1417         8649       141.70       864.90     Dec 02 18:33
at-spi-bus-laun      gabriel             1668            0            0            0            0     Dec 02 18:33
at-spi2-registr      gabriel             1752            8           24         0.80         2.40     Dec 02 18:33
bash                 gabriel            14379            0            0            0            0     Dec 02 20:21
code                 gabriel            14077            9           25         0.90         2.50     Dec 02 20:21
code                 gabriel            14079            0            0            0            0     Dec 02 20:21
code                 gabriel            14080            0            0            0            0     Dec 02 20:21
code                 gabriel            14128            1        10033         0.10      1003.30     Dec 02 20:21
code                 gabriel            14157           58           58         5.80         5.80     Dec 02 20:21
code                 gabriel            14251            0            0            0            0     Dec 02 20:21
code                 gabriel            14266         5734          513       573.40        51.30     Dec 02 20:21
code                 gabriel            14306           16           16         1.60         1.60     Dec 02 20:21
code                 gabriel            14360            0           62            0         6.20     Dec 02 20:21
code                 gabriel            14474            0            0            0            0     Dec 02 20:21
cpptools             gabriel            14768          455            0        45.50            0     Dec 02 20:21
dbus-daemon          gabriel             1476            0            0            0            0     Dec 02 18:33
dbus-daemon          gabriel             1673          134          269        13.40        26.90     Dec 02 18:33
dconf-service        gabriel             1779            0            0            0            0     Dec 02 18:33
evolution-addre      gabriel             1790            0            0            0            0     Dec 02 18:33
evolution-alarm      gabriel             1893            8           24         0.80         2.40     Dec 02 18:33
evolution-calen      gabriel             1777            0            0            0            0     Dec 02 18:33
evolution-sourc      gabriel             1769            0            0            0            0     Dec 02 18:33
firefox              gabriel             2852      2139270      1489309    213927.00    148930.90     Dec 02 18:33
gdm-x-session        gabriel             1560            0            0            0            0     Dec 02 18:33
gjs                  gabriel             1808            0            0            0            0     Dec 02 18:33
gnome-calculato      gabriel            30037            8           24         0.80         2.40     Dec 02 22:12
gnome-calendar       gabriel             2461            8           24         0.80         2.40     Dec 02 18:33
gnome-control-c      gabriel            30033            8           24         0.80         2.40     Dec 02 22:12
gnome-session-b      gabriel             1573            0            0            0            0     Dec 02 18:33
gnome-session-b      gabriel             1686            0            0            0            0     Dec 02 18:33
gnome-session-c      gabriel             1679            0            0            0            0     Dec 02 18:33
gnome-shell          gabriel             1700          775         4828        77.50       482.80     Dec 02 18:33
gnome-shell-cal      gabriel             1758            0            0            0            0     Dec 02 18:33
gnome-terminal-      gabriel            13905            8         9020         0.80       902.00     Dec 02 20:21
goa-daemon           gabriel             1529            0            0            0            0     Dec 02 18:33
goa-identity-se      gabriel             1540            0            0            0            0     Dec 02 18:33
gsd-a11y-settin      gabriel             1823            0            0            0            0     Dec 02 18:33
gsd-color            gabriel             1824           16           40         1.60         4.00     Dec 02 18:33
gsd-datetime         gabriel             1825            0            0            0            0     Dec 02 18:33
gsd-disk-utilit      gabriel             1903            0            0            0            0     Dec 02 18:33
gsd-housekeepin      gabriel             1827            0            0            0            0     Dec 02 18:33
gsd-keyboard         gabriel             1829            8           24         0.80         2.40     Dec 02 18:33
gsd-media-keys       gabriel             1831            8           24         0.80         2.40     Dec 02 18:33
gsd-power            gabriel             1832            8           24         0.80         2.40     Dec 02 18:33
```

Neste teste, é basicamente usado a opção “-c” , isto é, vai buscar processos com o nome/letra do argumento.Neste caso o argumento introduzido foi o “I.*”.

```
gabriel@gabriel-HP-ENVY-Laptop-14-eb0xxx:~/Desktop/SO/Projeto1$ ./rwstat.sh -c "I.*" 10
COMM                 USER                 PID        READB        WRITEB        RATER        RATEW             DATE
Isolated_Servic      gabriel           116708        36802           64      3680.20         6.40     Dec 02 23:38
Isolated_Web_Co      gabriel             3170           73           73         7.30         7.30     Dec 02 18:33
Isolated_Web_Co      gabriel            13224          152          152        15.20        15.20     Dec 02 20:08
Isolated_Web_Co      gabriel            13378            6            6         0.60          .60     Dec 02 20:09
Isolated_Web_Co      gabriel            13489            6            6         0.60          .60     Dec 02 20:10
Isolated_Web_Co      gabriel            13812            6            6         0.60          .60     Dec 02 20:16
Isolated_Web_Co      gabriel            18907           68           68         6.80         6.80     Dec 02 20:23
Isolated_Web_Co      gabriel            19832           44           44         4.40         4.40     Dec 02 20:40
Isolated_Web_Co      gabriel           116161           15           15         1.50         1.50     Dec 02 23:28
```

Neste teste, é basicamente usado a opção “-u” , isto é, vai buscar *user* que o utilizador usar no argumento.Neste caso foi introduzido “gabriel” .

```
gabriel@gabriel-HP-ENVY-Laptop-14-eb0xxx:~/Desktop/SO/Projeto1$ ./rwstat.sh -u gabriel 10
COMM                 USER                 PID         READB         WRITEB         RATER         RATEW             DATE
Isolated_Servic      gabriel            56138           839            30         83.90          3.00     Dec 02 22:51
Isolated_Web_Co      gabriel             3170           113           113         11.30         11.30     Dec 02 18:33
Isolated_Web_Co      gabriel            13224          2059          2059        205.90        205.90     Dec 02 20:08
Isolated_Web_Co      gabriel            13378             2             2          0.20           .20     Dec 02 20:09
Isolated_Web_Co      gabriel            13489             2             2          0.20           .20     Dec 02 20:10
Isolated_Web_Co      gabriel            13812            16            16          1.60          1.60     Dec 02 20:16
Isolated_Web_Co      gabriel            18907            39            39          3.90          3.90     Dec 02 20:23
Isolated_Web_Co      gabriel            19832            54            54          5.40          5.40     Dec 02 20:40
Privileged_Cont      gabriel             2967            20            20          2.00          2.00     Dec 02 18:33
RDD_Process          gabriel             3228             2             2          0.20           .20     Dec 02 18:33
Socket_Process       gabriel             2923             2             2          0.20           .20     Dec 02 18:33
Utility_Process      gabriel             3230             2             2          0.20           .20     Dec 02 18:33
WebExtensions        gabriel             3016            12            12          1.20          1.20     Dec 02 18:33
Web_Content          gabriel            57728             4             4          0.40           .40     Dec 02 22:53
Web_Content          gabriel            57777        474160            12      47416.00          1.20     Dec 02 22:55
Web_Content          gabriel            57844             6             6          0.60           .60     Dec 02 22:55
Xorg                 gabriel             1562         73210        258326       7321.00      25832.60     Dec 02 18:33
at-spi-bus-laun      gabriel             1668             0             0             0             0     Dec 02 18:33
at-spi2-registr      gabriel             1752             0          3024             0        302.40     Dec 02 18:33
bash                 gabriel            14379             0             0             0             0     Dec 02 20:21
code                 gabriel            14077             0          8064             0        806.40     Dec 02 20:21
code                 gabriel            14079             0             0             0             0     Dec 02 20:21
code                 gabriel            14080             0             0             0             0     Dec 02 20:21
code                 gabriel            14128             0             0             0             0     Dec 02 20:21
code                 gabriel            14157            58            58          5.80          5.80     Dec 02 20:21
code                 gabriel            14251             0             0             0             0     Dec 02 20:21
code                 gabriel            14266          5734           514        573.40         51.40     Dec 02 20:21
code                 gabriel            14306            16            16          1.60          1.60     Dec 02 20:21
code                 gabriel            14360             0            62             0          6.20     Dec 02 20:21
code                 gabriel            14474             0             0             0             0     Dec 02 20:21
cpptools             gabriel            14768           456             0         45.60             0     Dec 02 20:21
dbus-daemon          gabriel             1476             0             0             0             0     Dec 02 18:33
```

No seguinte teste, passamos a opção “-p”, sendo que esperados que só sejam listados o número de processo no terminal, ao correr o *script*, a opção “-m” serve só para aumentar o número de possibilidades de processos.

```
gabriel@gabriel-HP-ENVY-Laptop-14-eb0xxx:~/Desktop/SO/Projeto1$ ./rwstat.sh -m 70000 -p 6 10
COMM                 USER                 PID         READB         WRITEB         RATER         RATEW             DATE
Isolated_Servic      gabriel            86214             4             4          0.40           .40     Dec 02 23:12
Web_Content          gabriel            96689             2             2          0.20           .20     Dec 02 23:13
Web_Content          gabriel           104419             4             4          0.40           .40     Dec 02 23:15
Web_Content          gabriel           106004             4             4          0.40           .40     Dec 02 23:16
rwstat.sh            gabriel            71610             0             0             0             0     Dec 02 23:05
rwstat.sh            gabriel            73040             0             0             0             0     Dec 02 23:06
```

Datas “-e” é “-s”

No seguinte teste, passamos a opção “-e”, e de seguida um argumento, sendo que o esperado são processos só com data inferior em relação à data do processo.

```
gabriel@gabriel-HP-ENVY-Laptop-14-eb0xxx:~/Desktop/SO/Projeto1$ ./rwstat.sh -e "Dec 02 11:20" 10
COMM                 USER                  PID       READB       WRITEB        RATER        RATEW             DATE
Xorg                 gabriel              1559        1001        8037       100.10       803.70     Dec 02 11:17
at-spi-bus-laun      gabriel              1663           0           0            0            0     Dec 02 11:17
at-spi2-registr      gabriel              1742           8          24         0.80         2.40     Dec 02 11:17
dbus-daemon          gabriel              1473           0           0            0            0     Dec 02 11:17
dbus-daemon          gabriel              1668         201         401        20.10        40.10     Dec 02 11:17
dconf-service        gabriel              1769           0           0            0            0     Dec 02 11:17
evolution-addre      gabriel              1782           0           0            0            0     Dec 02 11:17
evolution-alarm      gabriel              1869           8          24         0.80         2.40     Dec 02 11:17
evolution-calen      gabriel              1767           0           0            0            0     Dec 02 11:17
evolution-sourc      gabriel              1757           0           0            0            0     Dec 02 11:17
gdm-x-session        gabriel              1557           0           0            0            0     Dec 02 11:17
gjs                  gabriel              1797           0           0            0            0     Dec 02 11:17
gnome-session-b      gabriel              1570           0           0            0            0     Dec 02 11:17
gnome-session-b      gabriel              1681           0           0            0            0     Dec 02 11:17
gnome-session-c      gabriel              1674           0           0            0            0     Dec 02 11:17
gnome-shell          gabriel              1695         424        4928        42.40       492.80     Dec 02 11:17
gnome-shell-cal      gabriel              1748           0           0            0            0     Dec 02 11:17
goa-daemon           gabriel              1526           0           0            0            0     Dec 02 11:17
goa-identity-se      gabriel              1537         128          48        12.80         4.80     Dec 02 11:17
gsd-a11y-settin      gabriel              1814           0           0            0            0     Dec 02 11:17
gsd-color            gabriel              1815           8          24         0.80         2.40     Dec 02 11:17
gsd-datetime         gabriel              1816           0           0            0            0     Dec 02 11:17
gsd-disk-utilit      gabriel              1878           0           0            0            0     Dec 02 11:17
gsd-housekeepin      gabriel              1817       16841           0      1684.10            0     Dec 02 11:17
gsd-keyboard         gabriel              1818           8          24         0.80         2.40     Dec 02 11:17
gsd-media-keys       gabriel              1821           8          24         0.80         2.40     Dec 02 11:17
gsd-power            gabriel              1822           8          24         0.80         2.40     Dec 02 11:17
gsd-print-notif      gabriel              1823           0           0            0            0     Dec 02 11:17
gsd-printer          gabriel              1923           0           0            0            0     Dec 02 11:17
gsd-rfkill           gabriel              1826           0           0            0            0     Dec 02 11:17
gsd-screensaver      gabriel              1827           0           0            0            0     Dec 02 11:17
gsd-sharing          gabriel              1828          24          40         2.40         4.00     Dec 02 11:17
gsd-smartcard        gabriel              1843           0           0            0            0     Dec 02 11:17
gsd-sound            gabriel              1846           0           0            0            0     Dec 02 11:17
gsd-usb-protect      gabriel              1847           0           0            0            0     Dec 02 11:17
gsd-wacom            gabriel              1848           8          24         0.80         2.40     Dec 02 11:17
gsd-wwan             gabriel              1849           0           0            0            0     Dec 02 11:17
gsd-xsettings        gabriel              1854           8          24         0.80         2.40     Dec 02 11:17
gvfs-afc-volume      gabriel              1513          40          80         4.00         8.00     Dec 02 11:17
gvfs-goa-volume      gabriel              1522          48          80         4.80         8.00     Dec 02 11:17
gvfs-gphoto2-vo      gabriel              1518          48          80         4.80         8.00     Dec 02 11:17
gvfs-mtp-volume      gabriel              1542          48          80         4.80         8.00     Dec 02 11:17
gvfs-udisks2-vo      gabriel              1505          48          80         4.80         8.00     Dec 02 11:17
gvfsd                gabriel              1480           0           0            0            0     Dec 02 11:17
gvfsd-fuse           gabriel              1486           0           0            0            0     Dec 02 11:17
gvfsd-metadata       gabriel              3084           0           0            0            0     Dec 02 11:18
gvfsd-trash          gabriel              1809          72         120         7.20        12.00     Dec 02 11:17
ibus-daemon          gabriel              1721           0           0            0            0     Dec 02 11:17
ibus-engine-sim      gabriel              1989           0           0            0            0     Dec 02 11:17
ibus-extension-      gabriel              1726           8          24         0.80         2.40     Dec 02 11:17
ibus-memconf         gabriel              1725           0           0            0            0     Dec 02 11:17
ibus-portal          gabriel              1730          72         120         7.20        12.00     Dec 02 11:17
ibus-x11             gabriel              1728           8          24         0.80         2.40     Dec 02 11:17
pulseaudio           gabriel              1467        1800        1800       180.00       180.00     Dec 02 11:17
snap-store           gabriel              1926           8          24         0.80         2.40     Dec 02 11:17
systemd              gabriel              1461     2014285       15966    201428.50      1596.60     Dec 02 11:17
tracker-miner-f      gabriel              1469           0           0            0            0     Dec 02 11:17
update-notifier      gabriel              3087           8          24         0.80         2.40     Dec 02 11:18
xdg-desktop-por      gabriel              2095          88         160         8.80        16.00     Dec 02 11:17
```

No seguinte teste, passamos a opção “-s”, e de seguida um argumento, sendo que o esperado são processos só com data superior em relação à data do processo.

```
gabriel@gabriel-HP-ENVY-Laptop-14-eb0xxx:~/Desktop/SO/Projeto1$ ./rwstat.sh -s "Dec 02 22:00" 10
COMM                 USER                  PID         READB        WRITEB         RATER         RATEW          DATE
Web_Content          gabriel             35309             4             4          0.40           .40  Dec 02 22:42
Web_Content          gabriel             35347             2             2          0.20           .20  Dec 02 22:42
Web_Content          gabriel             35389             4             4          0.40           .40  Dec 02 22:43
gnome-control-c      gabriel             35420             0             0             0             0  Dec 02 22:45
rwstat.sh            gabriel             35525      98113474        259549    9811347.40      25954.90  Dec 02 22:45
```

Nos seguintes testes, passamos às opção “-e” e “-s”, e de seguida um argumento à frente de cada opção, sendo que o esperado são processos só com data superior e inferior em relação à data do processo.

```
gabriel@gabriel-HP-ENVY-Laptop-14-eb0xxx:~/Desktop/SO/Projeto1$ ./rwstat.sh -s "Dec 02 19:20" 10
COMM                 USER                  PID         READB        WRITEB         RATER         RATEW          DATE
Isolated_Web_Co      gabriel             13224            10            10          1.00          1.00  Dec 02 20:08
Isolated_Web_Co      gabriel             13378             8             8          0.80           .80  Dec 02 20:09
Isolated_Web_Co      gabriel             13489             6             6          0.60           .60  Dec 02 20:10
Isolated_Web_Co      gabriel             13812             6             6          0.60           .60  Dec 02 20:16
Isolated_Web_Co      gabriel             18907            55            55          5.50          5.50  Dec 02 20:23
Isolated_Web_Co      gabriel             19832            54            54          5.40          5.40  Dec 02 20:40
Web_Content          gabriel             21836             2             2          0.20           .20  Dec 02 21:58
Web_Content          gabriel             21878             8             8          0.80           .80  Dec 02 22:01
Web_Content          gabriel             21939             2             2          0.20           .20  Dec 02 22:03
bash                 gabriel             14379             0             0             0             0  Dec 02 20:21
code                 gabriel             14077             0             0             0             0  Dec 02 20:21
code                 gabriel             14079             0             0             0             0  Dec 02 20:21
code                 gabriel             14080             0             0             0             0  Dec 02 20:21
code                 gabriel             14128             0             0             0             0  Dec 02 20:21
code                 gabriel             14157            58            58          5.80          5.80  Dec 02 20:21
code                 gabriel             14251             0             0             0             0  Dec 02 20:21
code                 gabriel             14266          5734           512        573.40         51.20  Dec 02 20:21
code                 gabriel             14306            16            16          1.60          1.60  Dec 02 20:21
code                 gabriel             14360             0            62             0          6.20  Dec 02 20:21
code                 gabriel             14474             0             0             0             0  Dec 02 20:21
cpptools             gabriel             14768           454             0         45.40             0  Dec 02 20:21
gnome-terminal-      gabriel             13905             0          9880             0        988.00  Dec 02 20:21
rwstat.sh            gabriel             26415      88441260        282269    8844126.00      28226.90  Dec 02 22:06
```

Segundo exemplo:

```
gabriel@gabriel-HP-ENVY-Laptop-14-eb0xxx:~/Desktop/SO/Projeto1$ ./rwstat.sh -s "Dec 02 11:20" -e "Dec 02 11:22" 10
COMM                 USER                  PID         READB        WRITEB         RATER         RATEW          DATE
gnome-calendar       gabriel              3299             0             0             0             0  Dec 02 11:21
```

Gama de pids “-m” e “-M”

No seguinte teste, usamos as opções “-m” e “-M” ,passamos-as com a informação que o gama de pids do início que é 3000 e a do fim é 5000, deste modo aparece uma mensagem com todas as informações impostas pelo utilizador

```
gabriel@gabriel-HP-ENVY-Laptop-14-eb0xxx:~/Desktop/SO/Projeto1$ ./rwstat.sh -m 3000 -M 5000 -c "d.*" 10
COMM                 USER                  PID         READB         WRITEB          RATER          RATEW             DATE
gnome-calendar       gabriel              3299             0              0              0              0     Dec 02 11:21
gvfsd-metadata       gabriel              3084             0              0              0              0     Dec 02 11:18
update-notifier      gabriel              3087             0              0              0              0     Dec 02 11:18
```

No seguinte teste , passamos a opção “-m” com a informação que o gama de pids mínima de 70000, deste modo aparece todas as informações superiores a esse número imposto pelo utilizador.

```
gabriel@gabriel-HP-ENVY-Laptop-14-eb0xxx:~/Desktop/SO/Projeto1$ ./rwstat.sh -m 70000 10
./rwstat.sh: linha 251: 317625-: erro de sintaxe: operando esperado (símbolo de erro é "-")
COMM                 USER                  PID         READB         WRITEB          RATER          RATEW             DATE
Isolated_Servic      gabriel             71579         37587             40        3758.70           4.00     Dec 02 23:05
Web_Content          gabriel             78991             0              0              0              0     Dec 02 23:07
Web_Content          gabriel             79290             2              2           0.20            .20     Dec 02 23:08
rwstat.sh            gabriel             71610             0              0              0              0     Dec 02 23:05
rwstat.sh            gabriel             73040             0              0              0              0     Dec 02 23:06
rwstat.sh            gabriel             74509             0              0              0              0     Dec 02 23:06
rwstat.sh            gabriel             79312      13297888         208347     1329788.80       20834.70     Dec 02 23:09
sleep                gabriel             73039             0              0              0              0     Dec 02 23:05
sleep                gabriel             74497             0              0              0              0     Dec 02 23:06
sleep                gabriel             75994             0              0              0              0     Dec 02 23:06
```

No seguinte teste , passamos a opção “-M” com a informação que o gama de pids máxima de 1500, deste modo aparece todas as informações inferiores a esse número imposto pelo utilizador.

```
gabriel@gabriel-HP-ENVY-Laptop-14-eb0xxx:~/Desktop/SO/Projeto1$ ./rwstat.sh -M 1500 10
COMM                 USER                  PID         READB         WRITEB          RATER          RATEW             DATE
dbus-daemon          gabriel              1476             0              0              0              0     Dec 02 18:33
gvfsd                gabriel              1480             0              0              0              0     Dec 02 18:33
gvfsd-fuse           gabriel              1485             0              0              0              0     Dec 02 18:33
pulseaudio           gabriel              1471          1980           1980         198.00         198.00     Dec 02 18:33
systemd              gabriel              1465             0              0              0              0     Dec 02 18:33
tracker-miner-f      gabriel              1473           168            280          16.80          28.00     Dec 02 18:33
```

No seguinte teste, usamos as opções “-m” e “-M”, passamos-as com a ordem trocada, deste modo aparece uma mensagem, de erro, e a função menu() é chamada.

```
gabriel@gabriel-HP-ENVY-Laptop-14-eb0xxx:~/Desktop/SO/Projeto1$ ./rwstat.sh -m 5000 -M 3000 -c "d.*" 10
O PID minimo é superior ao máximo, gama inválida

                                Menu
Opções de seleção de processos:

    -c      -> Seleção por uma expressão regular
    -u      -> Seleção pelo nome do utilizador
    -s      -> Seleção por um periodo temporal(data mínima)
    -e      -> Seleção por um periodo temporal(data máxima)
    -m      -> Seleção por gama de PID(Pid minimo)
    -M      -> Seleção de processos da tabela por gama de PID- Pid máximo
    -p      -> Seleção de número de processos a visualizar

Opção de ordenação da tabela:

    -w      -> Ordenação da tabela por valores do write(ratew)
    -r      -> Ordenação reversa
```

Conclusão

O script em questão, rwstat.sh, permite visualizar as estatísticas dos processos que estão em execução na nossa máquina, é mais que isso, tal como pedido, permite filtrar a informação através de opções passados no terminal no momento da execução do script.

este trabalho, serviu para elucidar em muito os nosso conhecimentos, primeiramente
aprendemos a trabalhar com arrays associativos, que são uma ferramenta poderosíssima e, que são bastante úteis, pois permite guardar informação de acordo com uma key, alargamos também os nosso conhecimentos em relação ao path das diretorias/pastas, pois tivemos de ir buscar várias
informações dentro de diretorias específicas dadas no guião, por último podemos dizer que aprendemos também a lidar com o acesso e permissões de certas e determinadas diretorias/pastas.

Ao longo da realização deste trabalho , foram aparecendo algumas dificuldades, sendo sinceros, no início estávamos ambos um pouco perdidos, porém depois de algum tempo a tentar perceber o que o trabalho realmente pedia. e como através de programação em bash, se conseguia responder as perguntas que o guirao proponha, foi-mais fácil implementar o código.
Neste ponto de realização do trabalho/projeto podemos afirmar que conseguimos alcançar todos os objetivos que o guião proponha, sendo desta forma muito satisfatória a realização deste trabalho prático.

Bibliografia

Para a realização deste trabalho prático, consultamos os slides disponíveis pelos professores na página da unidade curricular de Sistemas operativos bem como as aulas teóricas e práticas exercidas.

Auxílio de sites:

https://man.ex/bash
https://stackoverflow.com/
https://www.geeksforgeeks.org/awk-command-unixlinux-examples/
awk(1p) - Linux manual page
bash - What is the last argument of the previous command? - Unix & Linux Stack Exchange
bc command in Linux with examples - GeeksforGeeks
Comando ps no Linux (visualiza processos) [Guia Básico] - Certificação Linux
date(1) - Linux manual page