

# Taxas de Leitura/Escrita de Processos em bash

```
matilde@matilde-XPS-15-9520:~/SO/projeto$ ./rwstat.sh -m 3000 -w 1
```

COMM	USER	PID	READB	WRITEB	RATER	RATEW	DATE
rwstat.sh	matilde	362913	72644586	404966	72644586.00	404966.00	Dec 2 23:39
gnome-control-c	matilde	32852	3200	9560	3200.00	9560.00	Dec 2 17:39
Xwayland	matilde	3937	0	3072	0	3072.00	Dec 2 11:16
pulseaudio	matilde	3202	2691	2691	2691.00	2691.00	Dec 2 11:16
solaar	matilde	3611	5463	1760	5463.00	1760.00	Dec 2 11:16
Discord	matilde	27560	101392	930	101392.00	930.00	Dec 2 16:08
gnome-shell	matilde	3408	264	280	264.00	280.00	Dec 2 11:16
Discord	matilde	27488	130	130	130.00	130.00	Dec 2 16:08
gsd-sharing	matilde	3580	24	40	24.00	40.00	Dec 2 11:16
code	matilde	12466	0	31	0	31.00	Dec 2 11:48
TabNine	matilde	50152	0	26	0	26.00	Dec 2 20:37
code	matilde	50083	8	21	8.00	21.00	Dec 2 20:37
chrome	matilde	4967	16	16	16.00	16.00	Dec 2 11:16
code	matilde	12390	235	8	235.00	8.00	Dec 2 11:48
code	matilde	12447	8	8	8.00	8.00	Dec 2 11:48
Discord	matilde	27359	93574	2	93574.00	2.00	Dec 2 16:08
chrome	matilde	5016	110	0	110.00	0	Dec 2 11:16
pipewire-media-	matilde	3201	992	0	992.00	0	Dec 2 11:16

```
matilde@matilde-XPS-15-9520:~/SO/projeto$ ./rwstat.sh -c "c.*" -M 4000 -r 1
```

COMM	USER	PID	READB	WRITEB	RATER	RATEW	DATE
COMM	USER	PID	READB	WRITEB	RATER	RATEW	DATE
gnome-control-c	matilde	32852	3024	9072	3024.00	9072.00	Dec 2 17:39
Discord	matilde	27560	590	971	590.00	971.00	Dec 2 16:08
Discord	matilde	27488	135	135	135.00	135.00	Dec 2 16:08
Discord	matilde	27359	140365	3	140365.00	3.00	Dec 2 16:08
code	matilde	50083	8	21	8.00	21.00	Dec 2 20:37
code	matilde	12466	0	31	0	31.00	Dec 2 11:48
code	matilde	12447	8	8	8.00	8.00	Dec 2 11:48
chrome	matilde	6768	384	24	384.00	24.00	Dec 2 11:17
chrome	matilde	5016	11	11	11.00	11.00	Dec 2 11:16
chrome	matilde	4967	22359	70623	22359.00	70623.00	Dec 2 11:16
chrome	matilde	268503	50	45	50.00	45.00	Dec 2 23:00

```
matilde@matilde-XPS-15-9520:~/SO/projeto$ ./rwstat.sh -c "c.*" -M 4000000 -r 1
```

**Relatório Trabalho 01 - Turma P3**

**Prof. Guilherme Campos**

**2022/23**

Maria Rafaela Alves Abrunhosa, 107658  
Matilde Moital Portugal Sampaio Teixeira, 108193

## Índice

1) Introdução.....	3
2) Metodologia.....	4
a) Primeira parte- Validações dos argumentos de entrada.....	5
b) Segunda parte- Leituras realizadas antes e depois do sleep, listagem dos processos e obtenção das informações.....	9
i) Primeira leitura.....	10
ii) Segunda leitura.....	11
c) Terceira parte - Filtragens (opções introduzidas como argumentos de entrada).....	12
3) Demonstrações e Testes de Robustez.....	15
4) Conclusão.....	20

# Introdução

Na aula de Sistemas Operativos foi-nos proposto fazer um trabalho com o propósito de avaliar as taxas de leitura/escrita de processos em bash. Assim, desenvolvendo um script com o nome “rwstat.sh”, o objetivo era apresentar as informações de processos que estivessem a acontecer no computador, formatados numa tabela.

Este permitiria a visualização de alguns aspetos de processo: o comando que o executa, o user a que está ligado, o seu id enquanto processo, da data de início da execução, a quantidade de bytes lidos (em bytes por segundo) e escritos (também em bytes por segundo) e consoantemente estes últimos dois pontos as taxas de leitura e escrita dos mesmos.

Para realizar a leitura das taxas, o utilizador deste programa necessitaria de introduzir o número de segundos para determinar a taxa Input/Output de caracteres lidos e escritos, que seria calculada utilizando também a diferença entre o número de caracteres lidos e escritos e dividindo pelo número de segundos mencionado anteriormente.

Este programa também permite a filtragem consoante os argumentos selecionados, permitindo a visualização da tabela consoante isso. Ao introduzir (-c) a seleção dos processos a visualizar pode ser realizada através de uma expressão regular, ou seja, procura na palavra se existe o padrão que se introduziu como argumento da opção (-c).

Ao inserir (-s e -e), pretendemos fazer uma especificação do período temporal no qual se deu o início do processo, imprimindo, apenas, os processos que cuja data que obedecem à especificação de data mínima (-s) e/ou data máxima (-e). Outras seleções poderão ser, ainda, consoante o nome do utilizador (-u), o número de processos a visualizar (-p), ou segundo uma gama de pids (sendo -m a gama mínima e -M a gama máxima).

Existem opções para alterar a ordenação da tabela, sendo uma delas para inverter a ordem da tabela (-r) e outra (-w) para sortear os valores tendo por base os *write values* (número de bytes escritos).

Ao longo deste relatório, será explicada a metodologia utilizada para resolver este problema.

## Metodologia

Nesta parte do relatório, iremos abordar como foi executado o método utilizado para a resolução do problema enunciado na introdução.

Este trabalho foi dividido em 3 partes de acordo com a evolução da programação do código: na primeira parte, o objetivo principal foi validar os argumentos de entrada, na segunda parte, foi realizar duas leituras, uma antes e outra depois do sleep \$segundos, de forma a listar os processos e a obter as suas informações e na terceira parte, procuramos focarmo-nos nas filtragens, consoante as opções introduzidas como argumentos de entrada na execução do código.

## Primeira Parte

### Validações dos argumentos de entrada

De acordo com o referido anteriormente, o nosso objetivo primordial foi o código conseguir apresentar a tabela com todos os dados dos processos corretamente formatados, validando os dados.

Com esse intuito decidimos dividir o nosso raciocínio em três casos executando os seguintes processos:

#### Validações Básicas

A primeira validação a executar, é a de quando não são introduzidos argumentos de entrada.

```
if (($# == 0)); then
    echo "Please, insert some arguments"
    exit
fi
```

A segunda validação consiste em verificar se o número de segundos introduzido se encontra em formato numérico e se se trata de um número inteiro (positivo e maior que 0).

```
if [[ ${@: -1} =~ $re ]]; then
    if [[ ${@: -1} -ge 0 ]]; then
        segundos=${@: -1}
    fi
else
    echo "insert some seconds or the number you have chose is not a positive number"
    exit 1
fi
```

A terceira validação consiste em verificar se a string `$OPTARG` (argumento introduzido depois da opção) for null.

```

if [[ -z "$OPTARG" ]]; then # -z -> string is null, evaluates if the string has zero length
    arropt[$options]="null"
else
    arropt[$options]="${OPTARG}" # $OPTARG gets the value of the argument "linked" to the option
fi

```

## Validações de Opção - com recurso ao *case*

Ao entrar no *case*, consoante o comando seleccionado será validada a expressão inserida.

Assim, as verificações serão realizadas consoante a opção seleccionada, adoptando as validações necessárias para que estes argumentos sejam o mais corretos possíveis.

Deste modo a validação para o comando (-c e -u) irá verificar se a expressão inserida é nula, se o primeiro caracter da expressão inserida é um traço, e por fim se a expressão inserida não é numérica.

```

c) # seleção dos processos a visualizar através de uma expressão regular
expreg=${arropt['c']}
if [[ $expreg == 'null' || ${expreg:0:1} == "-" || ${expreg} =~ $re ]]; then
    echo "é necessário que a seleção de processos seja feita através de uma expressão regular"
    echo "o argumento de '-c' é 'null' ou inválido"
    exit 1
fi
;;

```

```

u) # seleção realizada pelo nome de utilizador
user=${arropt['u']}
if [[ $user == 'null' || ${user:0:1} == "-" || ${user} =~ $re ]]; then
    echo "o argumento de '-u' é 'null' ou inválido"
    exit 1
fi
;;

```

Ao ser inserido o comando (-s), este irá verificar se esta data não é nula, se o primeiro caracter da opção seleccionada é um traço e se se cinge ao padrão de datas (\$*timepattern*). Estas serão exatamente as mesmas verificações que serão aplicadas quando inserida a opção (-e).

```

case "$options" in
c) # seleção dos processos a visualizar através de uma expressão regular
    expreg=${arroppt['c']}
    if [[ $expreg == 'null' || ${expreg:0:1} == "-" || $expreg =~ $re ]]; then
        echo "é necessário que a seleção de processos seja feita através de uma expressão regular"
        echo "o argumento de '-c' é 'null' ou inválido"
        exit 1
    fi
    ;;
s) # especificação do período temporal - data mínima
    datemin=${arroppt['s']}
    if [[ $datemin == 'null' || ${datemin:0:1} == "-" || ! "${datemin}" =~ $timepattern ]]; then
        echo "o argumento de '-s' é 'null' ou inválido"
        exit 1
    fi
    start=$(date -d "$datemin" +%s);
    ;;
e) # especificação do período temporal - data máxima
    datemax=${arroppt['e']}
    if [[ $datemax == 'null' || ${datemax:0:1} == "-" || ! "${datemax}" =~ $timepattern ]]; then
        echo "o argumento de '-e' é 'null' ou inválido"
        exit 1
    fi
    end=$(date -d "$datemax" +%s);
    ;;

```

Quando selecionada a opção (-m e -M) irá verificar se a expressão inserida é nula, se o primeiro caracter da expressão inserida é um traço, se a expressão inserida não é numérica e finalmente se a gama (mínima se for introduzido -m, máxima se for introduzido -M) é maior ou igual a zero. Dentro do parâmetro -M, é ainda realizada outra verificação, para ver se a data máxima (-M) é menor que a data mínima (-m).

```

m) # gama de pids - gama mínima # FAZER AS GAMAS!!!
    gamamin=${arroppt['m']}
    if ! [[ $gamamin != 'null' || ${gamamin:0:1} != "-" || $gamamin =~ $re || $gamamin -ge 0 ]]; then
        echo "o argumento de '-m' é 'null', inválido ou menor que 0"
        exit 1
    fi
    ;;
M) # gama de pids - gama máxima
    gamamax=${arroppt['M']}
    if ! [[ $gamamin != 'null' || ${gamamin:0:1} != "-" || $gamamin =~ $re || $gamamin -ge 0 ]]; then
        if [ $gamamax -le $gamamin ]; then
            echo "gama máxima menor que gama mínima"
        fi
        echo "o argumento de '-m' é 'null', inválido ou menor que 0"
        exit 1
    fi
    ;;

```

Ao ser selecionada a opção (-p), este irá verificar se a expressão inserida é um número (se se enquadra no padrão regex, definido anteriormente como \$re).

```
p) # numero de processos a visualizar
numproc=${arrop['p']}
if ! [[ $numproc =~ $re ]]; then
    echo "o argumento de '-p' tem de ser um número"
    exit 1
fi
;;
```

A última verificação consistirá em verificar se nenhuma das opções (-c, -s, -e, -u, -m, -M, -p, -r, -w) foi selecionada mas alguma opção desconhecida foi introduzida, surgindo um erro e saindo do fluxo de execução.

```
*) # opção inválida
    echo "ERROR: Unknown option"
    exit 1
;;
esac
```

## Mais validações

Serão realizadas, por fim, mais duas verificações, que serão posteriormente utilizadas nas filtragens, -r e -w, respetivamente. Esta consistirá em verificar se o número de vezes em que foi selecionada cada uma das opções de ordenação (-r) ou (-w) é maior do que 1 ( $\$totalreverse > 1$ ), o que se acontecer devolve um erro, o mesmo será executado para ( $\$totalorder > 1$ ).

```
# more validations

if [[ $totalreverse -gt 1 ]]; then
    echo "ERROR: incompatible commands (-r > 1)"
    exit 1
fi

if [[ $totalorder -gt 1 ]]; then
    echo "ERROR: incompatible commands (-w > 1)"
    exit 1
fi
```



## Segunda Parte

### Leituras realizadas antes e depois do sleep, listagem dos processos e obtenção das informações

Como mencionado na metodologia, o segundo objetivo programado para a execução do código foi realizar duas leituras. Realizaram-se assim duas leituras, uma antes e outra depois do “sleep \$segundos” para que se pudesse calcular a diferença do número total de bytes de I/O que um processo leu/escreveu num intervalo de ‘s’ segundos bem como a taxa de leitura/escrita correspondente. São estas leituras que permitem ter acesso às informações de cada processo para posteriormente se listar cada processo com os seus dados associados.

Antes de se proceder a cada leitura é necessário aceder ao diretório do Linux **proc** através do comando “*cd /proc*”. Este diretório é um sistema de arquivos virtual criado durante a inicialização do sistema. Contém informações úteis sobre todos os processos que estão a correr no momento, processos ativos, e é, por isto, considerado o centro de informação. Informação esta que nos será útil na resolução deste problema.

No início do código foram inicializados 4 arrays associativos sendo que o *arropt* já foi explicado anteriormente. O array associativo *arrproc* será usado na segunda leitura e os arrays *arrread* e o *arrwrite* serão utilizados na primeira leitura. Um array associativo é uma estrutura de dados composta por um conjunto de elementos formados por uma chave e valor que serão armazenados. Neste caso, a chave será sempre o número do *pid*, *process ID*, e os valores associados serão diferentes consoante o nome do array e o que se pretende armazenar.

```
declare -A arrproc=()
declare -A arrread=()
declare -A arrwrite=()
declare -A arropt=()
```

## Primeira leitura

Na primeira leitura, após acessarmos ao diretório */proc*, filtramos todos os processos ativos para que nos apareçam os únicos cujo nome é um número que, por sinal, será o *pid* de cada um dos processos. Para ter acesso a cada um dos processos listados de */proc*, geramos um ciclo *for loop* que irá percorrer a lista dos processos sendo-lhe atribuído, a cada processo, o nome de *process*.

De seguida, verificamos se o *pid* existe através do comando “*\$process/status*” e lemos o *pid*, se tivermos permissão para a leitura dos dados, através de “*\$process/io*”.

Dentro do ciclo *for* e do *if*, utilizamos o pipe (“|”), para colocarmos o output da leitura dos dados do *pid* como input para os valores de *rchar* e *wchar*, onde serão guardados os números de bytes lidos e escritos, respetivamente.

Para cada valor de *rchar* e *wchar* armazenados, estes serão adicionados como valor aos arrays associativos *arrread* e *arrwrite*, por esta ordem, associados a uma chave, o *process ID*, do programa que está a ser lido.

```
# procurar e listar os processos

cd /proc
for process in $(ls | grep -E '^[0-9]+$'); do
    if [[ -r $process/status && -r $process/io ]]; then
        pid=$(cat $process/status | grep -w Pid | tr -dc '0-9' )
        #printf "%d -> o valor do proc (pid) \n" "$pid"

        rchar=$(cat $process/io | grep rchar | grep -o -E '[0-9]+') # "|" pipe chains commands together -> takes the output from one command and feeds it t
        wchar=$(cat $process/io | grep wchar | grep -o -E '[0-9]+')

        #printf "%d -> o valor do rchar \n" "$rchar"
        #printf "%d -> o valor do wchar \n" "$wchar"

        if [[ $rchar == 0 && $wchar == 0 ]]; then
            continue
        else
            arrread[$pid]=$rchar
            arrwrite[$pid]=$wchar
        fi
    fi
done
cd
```

Depois da primeira leitura ser efetuada, esperemos pelo comando “*sleep \$segundos*” que parará o *script* pela quantidade de segundos passados como argumento inicial.

## Segunda leitura

Passando para a segunda leitura, utilizamos o mesmo processo da primeira leitura para entrarmos no diretório e filtrarmos os processos ativos como também, para verificar a existência do *pid* e a sua posterior leitura.

É nesta leitura que obtemos e guardamos o nome do processo (*comm*), substituindo os espaços por “\_”, o utilizador do processo (*userproc*) e o período temporal em que o mesmo é iniciado (*data\_2*).

Voltamos também a calcular o número de bytes lidos e escritos, pelo mesmo processo, mas desta vez guardamos numa segunda variável, *rchar\_2* e *wchar\_2*, que são os valores obtidos após se esperar o número de segundos desejado inicialmente.

Na impressão da tabela, os valores dos bytes lidos e escritos representados serão a diferença entre o número de bytes após os ‘s’ segundos e o número de bytes iniciais (“*\$rchar\_2 - \${arrread[\$pid]}*”) e “*\$wchar\_2 - \${arrwrite[\$pid]}*”). Para aceder ao valor dos *rchar* e *wchar* iniciais é preciso aceder aos arrays associativos *arrread* e *arrwrite* onde a chave será o *pid* do processo em que estamos.

Por fim, para efetuar o cálculo das taxas de leitura e escrita, basta dividir o valor da diferença calculada pelo número nos ‘s’ segundos esperados. Os valores das taxas vão ter, no máximo, duas casas decimais (“*scale=2*”) e, por default, os números que forem “.(...)”, sem ‘0’, vão passar a tê-lo através da substituição da expressão “*#./0.*”.

```
cd /proc
for process in $(ls | grep -E '^[0-9]+$'); do
  if [[ -r $process/status && -r $process/io ]]; then
    pid=$(cat $process/status | grep -w Pid | tr -dc '0-9' )
    #printf "%d -> o valor do proc (pid) 2\n" "$pid"
    comm=$(cat $process/comm | tr " " "_") # ir buscar o comm,e retirar os espaços e substituir por '_' nos comm's com 2 nomes
    #printf "%s -> o valor do comm 2\n" "$comm"
    userproc=$(ps -o user= -p $pid)
    # printf "%s -> o valor do user \n" "$userproc"
    data=$(LC_ALL=EN us.utf8 ls -ld /proc/$process | awk '{print $6 " " $7 " " $8}')
    data_2=$(date -d "$data" +%s)
```

```
rchar_2=$(cat $process/io | grep rchar | grep -o -E '[0-9]+')
wchar_2=$(cat $process/io | grep wchar | grep -o -E '[0-9]+')

subread=$((rchar_2 - ${arrread[$pid]}))
#echo "calcula $subread"
#printf "eu vou comer\n"
subwrite=$((wchar_2 - ${arrwrite[$pid]}))
#echo "calcula 2 $subwrite"
```

Depois de se obter todas as informações desejadas e de se ter todos os cálculos efetuados, adiciona-se todas estas ao array associativo *arrproc*, já com a formatação correta a partir do *printf*, onde a chave será, mais uma vez, o *pid* e o seu valor associado, as informações adquiridas.

```
arrproc[$pid]=$(printf "%-20s %-13s %8s %11s %11s %11s %13s %13s %13s\n" "$comm" "$userproc" "$pid" "$subread" "$subwrite" "$rater" "$ratew" "$data")
```

## Terceira Parte

### Filtragens - opções introduzidas como argumentos de entrada

De acordo com as opções possíveis de serem selecionadas (-c, -s, -e, -u, -m, -M, -p, -r e -w), os processos irão ser filtrados segundo a opção selecionada tal como foi explicado na introdução.

Algumas destas filtragens serão realizadas na segunda leitura, enquanto que outras filtragens só serão realizadas na impressão da tabela.

As filtragens a ocorrer na segunda leitura serão as que se centram nos comandos (-u, -c, -s, -e, -m, -M).

Se a opção selecionada for -u, este vai verificar se a opção -u (-v informa a shell para rodar em *verbose mode* (localiza o processo que cria um erro)) e se o que está na posição do array de opções com 'u' como chave (user introduzido pelo utilizador), não for igual ao \$userproc (o user do processo que estamos a analisar) este vai continuar a executar sem dar print a esse processo. Este caso irá ser igual com a opção -c.

```
if [[ -v arropt[u] && ! ${arropt['u']} == $userproc ]]; then # -v tells the shell to run in verbose mode -> is useful
# in locating the line of the script that as created an error
|   continue # se não existir o user escolhido não adiciona à lista
fi
```

```
if [[ -v arropt[c] && ! $comm =~ ${arropt['c']} ]]; then # escolhe quais "$comm" se adequam ao padrão inserido
|   continue
fi
```

Na opção -s, este irá ver se existe algum erro no array de opções cuja chave é 's', como feito com o comando -u, e se tal acontecer, este irá verificar se \$data\_2 (data de início do processo) é menor que a data de início introduzida pelo utilizador como argumento de -s. Se forem diferentes, irá continuar não executando o print dos elementos do array que tenham essa informação. Com o comando (-e) este vai fazer exatamente o mesmo, mas comparando agora o \$data\_2 com o \$end, vendo se esta primeira é maior que a introduzida pelo utilizador (\$end).

```
if [[ -v arropt[s] ]]; then
|   if [[ "$data_2" -lt "$start" ]]; then
|       continue
|   fi
fi

if [[ -v arropt[e] ]]; then
|   if [[ "$data_2" -gt "$end" ]]; then
|       continue
|   fi
fi
```

Para a filtragem das gamas de pids (-m seleciona a gama mínima e -M seleciona gama máxima), este irá ver se existe algum erro no array de opções cuja chave é m, e se tal acontecer este irá verificar se \$pid (lido na segunda leitura) é menor que a gama mínima selecionada pelo utilizador, se isso acontecer, ele não vai dar print ao processo correspondente aos elementos do array com essa informação. Em seguida, vai fazer exatamente o mesmo com -M, desta vez verificando se \$pid vai ser maior que a gama máxima e se for, não vai dar print a esse processo.

```
if [[ -v arropt[m] ]]; then
    if [[ "$pid" -lt "$gamamin" ]]; then
        continue
    fi
fi

if [[ -v arropt[M] ]]; then
    if [[ "$pid" -gt "$gamamax" ]]; then
        continue
    fi
fi
```

Na validação da opção (-p), o valor do *p* será, por default, igual ao tamanho do array onde estão armazenados todos os processos ("*p=\${#arrproc[@]}*"). Se for selecionada a opção (-p) e introduzido um valor para o mesmo, o *p* passa a ter esse valor que ficou armazenado no *arropt* de chave '*p*' ("*p=\${arropt['p']}*").

```
p) # numero de processos a visualizar
numproc=${arropt['p']}
if ! [[ $numproc =~ $re ]]; then
    echo "o argumento de '-p' tem de ser um número"
    exit 1
fi
;;
```

Além das filtragens realizadas na segunda leitura, irão ser realizadas mais duas verificações, executadas pelos comandos (-r e -w), como já estipulado no case, se a opção escolhida for (-r), *sort\_reverse=1*, este irá percorrer os diversos ifs na zona dos prints, e imprimir inversamente usando a opção (-r). Quanto à opção (-w), este irá ordenar com o mesmo princípio que no (-r), baseando-se na ordem atribuída (-f -r) quando o reverse está inativo. Quando o reverse e o sort nos write values estiverem os dois ativos, a impressão estará em ordem pelos write values mas invertidamente.

```
r) # sort reverse
((totalreverse++))
sort_reverse=1
;;

w) # sort nos valores do write
((totalorder++))
order=5
;;
```

```

if ! [[ -v arropt[p] ]]; then
|   p=${#arrproc[@]} # por default o valor de p é igual ao tamanho do array de todos os processos
|   ## -> indica o tamanho do array
else
|   p=${arropt['p']} # numero de processos inserido
fi

printf "%-20s %-18s %-1s %12s %12s %10s %10s %10s %10s %6s %-1s %-1s \n" "COMM" "USER" "PID" "READB" "WRITEB" "RATER" "RATEW" "DATE"

# printf '%s \n' "${arrproc[@]}" | head -n $p

if [[ $order -ne 1 && $sort_reverse -eq 0 ]]; then
|   printf "%s \n" "${arrproc[@]}" | sort -k${order}rn | head -n $p
elif [[ $order -ne 1 && $sort_reverse -eq 1 ]]; then
|   printf "%s \n" "${arrproc[@]}" | sort -k${order}n | head -n $p
elif [[ $order -eq 1 && $sort_reverse -eq 1 ]]; then
|   printf "%s \n" "${arrproc[@]}" | sort -k${order} -f -r | head -n $p
else
|   printf "%s \n" "${arrproc[@]}" | sort -k${order} -f | head -n $p
fi

```

## Demonstrações e Testes de Robustez

De modo a que se testasse o script, este foi corrido com diversos parâmetros. Quando corrido sem qualquer parâmetro, a mensagem que aparece no terminal é *"Please, insert some arguments"*.

```
matilde@matilde-XPS-15-9520:~/S0/projeto$ ./rwstat.sh
Please, insert some arguments
```

Apenas quando é introduzido um argumento de entrada (segundos) é que o script é executado.

```
matilde@matilde-XPS-15-9520:~/S0/projeto$ ./rwstat.sh 3
COMM      USER      PID      READB      WRITEB      RATER      RATEW      DATE
chrome    matilde    25602     5           0           1.66       0          Dec 2 15:55
chrome    matilde    4967      65          8753        21.66      2917.66    Dec 2 11:16
chrome    matilde    5016      4396        66128       1465.33    22042.66   Dec 2 11:16
chrome    matilde    6768      3           3           1.00       1.00       Dec 2 11:17
code      matilde    12292     2           2           0.66       0.66       Dec 2 11:48
code      matilde    12331     4           4           1.33       1.33       Dec 2 11:48
code      matilde    12390     52          39          17.33      13.00      Dec 2 11:48
code      matilde    12447     8           8           2.66       2.66       Dec 2 11:48
code      matilde    12466     0           31          0          10.33      Dec 2 11:48
code      matilde    50083     465         81          155.00     27.00      Dec 2 20:37
Discord   matilde    27359     233676      6           77892.00   2.00       Dec 2 16:08
Discord   matilde    27488     240         240         80.00      80.00      Dec 2 16:08
Discord   matilde    27560     101113      1430        33704.33   476.66     Dec 2 16:08
gnome-control-c matilde    32852     5456        16328       1818.66    5442.66    Dec 2 17:39
gnome-shell matilde    3408      408         424         136.00     141.33     Dec 2 11:16
gsd-sharing matilde    3580      24          40          8.00       13.33      Dec 2 11:16
pipewire-media- matilde    3201      928         0           309.33     0          Dec 2 11:16
pulseaudio matilde    3202      4963        4980        1654.33    1660.00    Dec 2 11:16
rwstat.sh matilde    220104    21373518    353262      7124506.00 117754.00   Dec 2 21:51
solaar    matilde    3611      7667        2480        2555.66    826.66     Dec 2 11:16
TabNine   matilde    50152     47          86          15.66      28.66      Dec 2 20:37
TabNine   matilde    50213     47          457         15.66      152.33     Dec 2 20:37
WD-TabNine matilde    45045     3485428     0           1161809.33 0          Dec 2 20:01
Xwayland  matilde    3937      0           4332        0          1444.00    Dec 2 11:16
```

Ao executar a operação `-c`, este atua ao selecionar somente os processos que contém no seu nome o argumento inserido após a seleção do comando `-c`. Note-se que se não for passado um argumento para o tempo, este programa transmitirá uma mensagem de erro.

```
matilde@matilde-XPS-15-9520:~/S0/projeto$ ./rwstat.sh -c "d.*"
insert some seconds or the number you have chose is not a positive number
matilde@matilde-XPS-15-9520:~/S0/projeto$ ./rwstat.sh -c "d.*" 3
COMM      USER      PID      READB      WRITEB      RATER      RATEW      DATE
code      matilde    12390     8           21          2.66       7.00       Dec 2 11:48
code      matilde    12447     8           8           2.66       2.66       Dec 2 11:48
code      matilde    12466     0           31          0          10.33      Dec 2 11:48
code      matilde    50083     42          29          14.00      9.66       Dec 2 20:37
Discord   matilde    27359     186935      5           62311.66   1.66       Dec 2 16:08
Discord   matilde    27488     215         215         71.66      71.66      Dec 2 16:08
Discord   matilde    27560     874         1636        291.33     545.33     Dec 2 16:08
gsd-sharing matilde    3580      24          40          8.00       13.33      Dec 2 11:16
pipewire-media- matilde    3201      960         0           320.00     0          Dec 2 11:16
pulseaudio matilde    3202      4873        4873        1624.33    1624.33    Dec 2 11:16
TabNine-deep-lo matilde    7679      0           39          0          13.00      Dec 2 11:18
Xwayland  matilde    3937      0           4556        0          1518.66    Dec 2 11:16
matilde@matilde-XPS-15-9520:~/S0/projeto$
```

Com os comandos (-s) e (-e) conseguimos filtrar os processos pelas datas de início e de fim, respectivamente. Esta opção irá filtrar os processos pelo momento em que começaram, cingindo-se ao intervalo de tempo (-s a -e). Note-se que quando introduzida apenas a data de início (-s) este irá listar apenas os processos que tenham ocorrido após esta data. O oposto irá acontecer com (-e), só irá listar os processos cuja data seja anterior à introduzida pelo utilizador.

Caso não haja processos nesse intervalo de tempo, então deverá aparecer no terminal somente o cabeçalho, apresentando assim a lista vazia.

```
matilde@matilde-XPS-15-9520:~/SO/projeto$ ./rwstat.sh -s "Dec 02 16:00" 3
COMM      USER      PID      READB      WRITEB      RATER      RATEW      DATE
code      matilde    50083     8           21          2.66       7.00       Dec 2 20:37
Discord    matilde    27359     187093      5           62364.33   1.66       Dec 2 16:08
Discord    matilde    27488     235         235         78.33      78.33      Dec 2 16:08
Discord    matilde    27560     101370      1416        33790.00   472.00     Dec 2 16:08
gnome-control-c matilde    32852     5168        15536       1722.66    5178.66    Dec 2 17:39
rwstat.sh  matilde    253945    23727571    347590      7909190.33 115863.33  Dec 2 22:56
TabNine    matilde    50152     0           39          0          13.00      Dec 2 20:37
TabNine    matilde    50213     372         3807        124.00     1269.00    Dec 2 20:37
WD-TabNine matilde    228962    3467947     0           1155982.33 0          Dec 2 21:59
matilde@matilde-XPS-15-9520:~/SO/projeto$ ./rwstat.sh -e "Dec 02 16:00" 3
COMM      USER      PID      READB      WRITEB      RATER      RATEW      DATE
chrome    matilde    25602     6           1           2.00       0.33       Dec 2 15:55
chrome    matilde    4967      152         12372       50.66      4124.00    Dec 2 11:16
chrome    matilde    5016      110         403176      36.66      134392.00  Dec 2 11:16
chrome    matilde    6768      4582        6           1527.33    2.00       Dec 2 11:17
code      matilde    12390     21          8           7.00       2.66       Dec 2 11:48
code      matilde    12447     8           8           2.66       2.66       Dec 2 11:48
code      matilde    12466     0           31          0          10.33      Dec 2 11:48
gnome-shell matilde    3408      408         424         136.00     141.33     Dec 2 11:16
gsd-sharing matilde    3580      24          40          8.00       13.33      Dec 2 11:16
pipewire-media- matilde    3201      1024        0           341.33     0          Dec 2 11:16
pulseaudio matilde    3202      5248        5248        1749.33    1749.33    Dec 2 11:16
solaar    matilde    3611      8577        2760        2859.00    920.00     Dec 2 11:16
Xwayland  matilde    3937      0           4812        0          1604.00    Dec 2 11:16
matilde@matilde-XPS-15-9520:~/SO/projeto$ ./rwstat.sh -s "Dec 02 11:48" -e "Dec 02 16:08" 2
COMM      USER      PID      READB      WRITEB      RATER      RATEW      DATE
code      matilde    12390     21          8           10.50      4.00       Dec 2 11:48
code      matilde    12447     8           8           4.00       4.00       Dec 2 11:48
code      matilde    12466     0           31          0          15.50      Dec 2 11:48
Discord    matilde    27359     140320      4           70160.00   2.00       Dec 2 16:08
Discord    matilde    27488     181         177         90.50      88.50      Dec 2 16:08
Discord    matilde    27560     802         1182        401.00     591.00     Dec 2 16:08
matilde@matilde-XPS-15-9520:~/SO/projeto$ ./rwstat.sh -s "Dec 02 22:00" -e "Dec 02 22:01" 2
COMM      USER      PID      READB      WRITEB      RATER      RATEW      DATE
```

Ao executar o comando -p, vai ser impresso no terminal o número de processos representado pelo argumento introduzido, tendo este que ser um número inteiro maior que 0.

```
matilde@matilde-XPS-15-9520:~/SO/projeto$ ./rwstat.sh -p 5 1
COMM      USER      PID      READB      WRITEB      RATER      RATEW      DATE
chrome    matilde    188649    32          0           32.00      0          Dec 2 21:47
chrome    matilde    25602     1           1           1.00       1.00       Dec 2 15:55
chrome    matilde    49143     1530        0           1530.00    0          Dec 2 20:23
chrome    matilde    4967      6949248     53370      6949248.00 53370.00   Dec 2 11:16
chrome    matilde    4988      0           32          0          32.00      Dec 2 11:17
matilde@matilde-XPS-15-9520:~/SO/projeto$ ./rwstat.sh -p 2 1
COMM      USER      PID      READB      WRITEB      RATER      RATEW      DATE
chrome    matilde    25602     29          173         29.00      173.00     Dec 2 15:55
chrome    matilde    4967      49099       66783       49099.00   66783.00   Dec 2 11:16
matilde@matilde-XPS-15-9520:~/SO/projeto$ ./rwstat.sh -p 8 1
COMM      USER      PID      READB      WRITEB      RATER      RATEW      DATE
chrome    matilde    25602     42          42          42.00      42.00      Dec 2 15:55
chrome    matilde    4967      68688       91984       68688.00   91984.00   Dec 2 11:16
chrome    matilde    5016      15          15          15.00      15.00      Dec 2 11:16
chrome    matilde    6768      254         38          254.00     38.00      Dec 2 11:17
code      matilde    12390     272         81          272.00     81.00      Dec 2 11:48
code      matilde    12447     8           8           8.00       8.00       Dec 2 11:48
code      matilde    12466     0           31          0          31.00      Dec 2 11:48
code      matilde    50083     4261        475         4261.00    475.00     Dec 2 20:37
```



Foram também desenvolvidas opções de ordenamento de processos, tais como a filtragem pela gama de pids (-m e -M).

A opção -m irá filtrar tendo como mínimo o argumento inserido pelo utilizador, enquanto que -M filtra tendo como máximo o argumento introduzido pelo utilizador.

Ao ser executada a opção -m e -M, ao mesmo tempo, verifica se que são somente listados os processos cujos pids se encontram na gama apresentada.

Quando introduzido um pid que não esteja na tabela, o programa irá somente mostrar o cabeçalho da tabela.

```
matilde@matilde-XPS-15-9520:~/50/projeto$ ./rwstat.sh -m 3000 3
COMM      USER      PID      READB      WRITEB      RATER      RATEW      DATE
chrome    matilde    268503    6           1           2.00        0.33      Dec 2 23:00
chrome    matilde    4967      180          80673        60.00       26891.00   Dec 2 11:16
chrome    matilde    5016      110          428300       36.66       142766.66  Dec 2 11:16
chrome    matilde    6768      4585         9           1528.33      3.00       Dec 2 11:17
code      matilde    12331     5            0           1.66         0          Dec 2 11:48
code      matilde    12390     21           8           7.00         2.66       Dec 2 11:48
code      matilde    50083     452          68          150.66       22.66      Dec 2 20:37
Discord   matilde    27359     233935       5           77978.33     1.66       Dec 2 16:08
Discord   matilde    27488     238          238          79.33        79.33      Dec 2 16:08
Discord   matilde    27560     101684       1821        33894.66     607.00     Dec 2 16:08
gnome-control-c matilde    32852     5544         16632       1848.00     5544.00    Dec 2 17:39
gnome-shell matilde    3408      408          424          136.00       141.33     Dec 2 11:16
gsd-sharing matilde    3580      24           40           8.00         13.33      Dec 2 11:16
pipewire-media- matilde    3201      1056         0           352.00        0          Dec 2 11:16
pulseaudio matilde    3202      5383         5383        1794.33     1794.33    Dec 2 11:16
rwstat.sh matilde    311937    32608206     368664      10869402.00 122888.00  Dec 2 23:10
solaar    matilde    3611      8391         2700        2797.00     900.00     Dec 2 11:16
TabNine   matilde    50152     47           86           15.66        28.66      Dec 2 20:37
TabNine   matilde    50213     47           457          15.66       152.33     Dec 2 20:37
WD-TabNine matilde    228962    3513082      0           1171027.33   0          Dec 2 21:59
Xwayland  matilde    3937      0            4652         0           1550.66     Dec 2 11:16

matilde@matilde-XPS-15-9520:~/50/projeto$ ./rwstat.sh -M 6000 3
COMM      USER      PID      READB      WRITEB      RATER      RATEW      DATE
chrome    matilde    4967      25          26           8.33        8.66      Dec 2 11:16
chrome    matilde    5016      111          1           37.00        0.33      Dec 2 11:16
gnome-shell matilde    3408      384          384          128.00       128.00     Dec 2 11:16
pipewire-media- matilde    3201      1056         0           352.00        0          Dec 2 11:16
pulseaudio matilde    3202      5253         5245        1751.00     1748.33    Dec 2 11:16
solaar    matilde    3611      8859         2860        2953.00     953.33     Dec 2 11:16
Xwayland  matilde    3937      0            5100         0           1700.00     Dec 2 11:16

matilde@matilde-XPS-15-9520:~/50/projeto$ ./rwstat.sh -m 3000 -M 6000 3
COMM      USER      PID      READB      WRITEB      RATER      RATEW      DATE
chrome    matilde    4967      27          27           9.00         9.00      Dec 2 11:16
chrome    matilde    5016      110          25124        36.66       8374.66    Dec 2 11:16
gnome-shell matilde    3408      384          384          128.00       128.00     Dec 2 11:16
gsd-housekeepin matilde    3563      15686         0           5228.66        0          Dec 2 11:16
pipewire-media- matilde    3201      1024         0           341.33        0          Dec 2 11:16
pulseaudio matilde    3202      4831         4831        1610.33     1610.33    Dec 2 11:16
solaar    matilde    3611      8074         2600        2691.33     866.66     Dec 2 11:16
Xwayland  matilde    3937      0            4588         0           1529.33     Dec 2 11:16

matilde@matilde-XPS-15-9520:~/50/projeto$ ./rwstat.sh -M 60 3
COMM      USER      PID      READB      WRITEB      RATER      RATEW      DATE
```

Ao ser executada a opção (-u), este irá filtrar somente pelos processos que tenham esse user. Caso o utilizador introduzido como parâmetro não tenha um processo associado, no terminal irá somente aparecer o cabeçalho da tabela.

```
matilde@matilde-XPS-15-9520:~/SO/projeto$ ./rwstat.sh -u matilde 2
COMM      USER      PID      READB      WRITEB      RATER      RATEW      DATE
chrome    matilde    188649    32          0          16.00      0          Dec 2 21:47
chrome    matilde    268503    62          134         31.00      67.00      Dec 2 23:00
chrome    matilde    4967      41056       17317       20528.00   8658.50    Dec 2 11:16
chrome    matilde    5016      246         8           123.00     4.00       Dec 2 11:16
chrome    matilde    6768      4950        22          2475.00    11.00      Dec 2 11:17
code      matilde    12390     21          8           10.50      4.00       Dec 2 11:48
code      matilde    50083     452         68          226.00     34.00      Dec 2 20:37
Discord   matilde    27359     140355      3           70177.50   1.50       Dec 2 16:08
Discord   matilde    27488     191         186         95.50      93.00      Dec 2 16:08
Discord   matilde    27499     33          0           16.50      0          Dec 2 16:08
Discord   matilde    27560     101060      1219        50530.00   609.50     Dec 2 16:08
gnome-control-c matilde    32852     4336       13000       2168.00    6500.00    Dec 2 17:39
gnome-shell matilde    3408      288         288         144.00     144.00     Dec 2 11:16
pipewire-media-pulseaudio matilde    3201      992         0           496.00     0          Dec 2 11:16
pulseaudio matilde    3202      3818        3818        1909.00    1909.00    Dec 2 11:16
rwstat.sh matilde    327887    42112460    377741      21056230.00 188870.50 Dec 2 23:19
solaar    matilde    3611      7426        2400        3713.00    1200.00    Dec 2 11:16
TabNine   matilde    50152     47          73          23.50      36.50      Dec 2 20:37
TabNine   matilde    50213     47          457         23.50      228.50     Dec 2 20:37
Xwayland  matilde    3937      0           4172        0          2086.00    Dec 2 11:16
matilde@matilde-XPS-15-9520:~/SO/projeto$ ./rwstat.sh -u rafaela 2
COMM      USER      PID      READB      WRITEB      RATER      RATEW      DATE
```

A opção (-w) executa uma filtragem pela taxa de escrita (-w), ordenando por forma a que os processos com menor taxa de escrita sejam listados primeiro.

Caso seja executada a opção (-r), a tabela será ordenada de forma inversa à ordem inicial.

```
matilde@matilde-XPS-15-9520:~/SO/projeto$ ./rwstat.sh 2
COMM      USER      PID      READB      WRITEB      RATER      RATEW      DATE
chrome    matilde    4967      114         76519       57.00      38259.50   Dec 2 11:16
chrome    matilde    5016      244         6           122.00     3.00       Dec 2 11:16
chrome    matilde    6768      104         8           52.00      4.00       Dec 2 11:17
code      matilde    12331     5           0           2.50       0          Dec 2 11:48
code      matilde    12390     235         8           117.50     4.00       Dec 2 11:48
code      matilde    12447     8           8           4.00       4.00       Dec 2 11:48
code      matilde    12466     0           31          0          15.50      Dec 2 11:48
code      matilde    50083     452         68          226.00     34.00      Dec 2 20:37
Discord   matilde    27359     140359      3           70179.50   1.50       Dec 2 16:08
Discord   matilde    27488     192         192         96.00      96.00      Dec 2 16:08
Discord   matilde    27560     673         1054        336.50     527.00     Dec 2 16:08
gnome-control-c matilde    32852     4320       12960       2160.00    6480.00    Dec 2 17:39
gnome-shell matilde    3408      336         336         168.00     168.00     Dec 2 11:16
gsd-color matilde    3559      8           16          4.00       8.00       Dec 2 11:16
pipewire-media-pulseaudio matilde    3201      992         0           496.00     0          Dec 2 11:16
pulseaudio matilde    3202      3849        3849        1924.50    1924.50    Dec 2 11:16
rwstat.sh matilde    335749    49679180    385066      24839590.00 192533.00 Dec 2 23:23
solaar    matilde    3611      7454        2400        3727.00    1200.00    Dec 2 11:16
TabNine-deep-lo matilde    7679      4           4           2.00       2.00       Dec 2 11:18
TabNine   matilde    50152     47          73          23.50      36.50      Dec 2 20:37
TabNine   matilde    50213     47          457         23.50      228.50     Dec 2 20:37
WD-TabNine matilde    228962    3519770     0           1759885.00 0          Dec 2 21:59
Xwayland  matilde    3937      0           4236        0          2118.00    Dec 2 11:16
matilde@matilde-XPS-15-9520:~/SO/projeto$ ./rwstat.sh -r 2
COMM      USER      PID      READB      WRITEB      RATER      RATEW      DATE
Xwayland  matilde    3937      0           14692       0          7346.00    Dec 2 11:16
TabNine   matilde    50213     47          457         23.50      228.50     Dec 2 20:37
TabNine   matilde    50152     47          73          23.50      36.50      Dec 2 20:37
solaar    matilde    3611      8715        2380        4357.50    1190.00    Dec 2 11:16
rwstat.sh matilde    340229    51870476    385978      25935238.00 192989.00 Dec 2 23:23
pulseaudio matilde    3202      3891        3891        1945.50    1945.50    Dec 2 11:16
pipewire-media-pulseaudio matilde    3201      992         0           496.00     0          Dec 2 11:16
gnome-shell matilde    3408      16784       12148       8392.00    6074.00    Dec 2 11:16
gnome-control-c matilde    32852     4368       13104       2184.00    6552.00    Dec 2 17:39
Discord   matilde    27560     101672      1223        50836.00   611.50     Dec 2 16:08
Discord   matilde    27499     406         0           203.00     0          Dec 2 16:08
Discord   matilde    27488     199         194         99.50      97.00      Dec 2 16:08
Discord   matilde    27359     140356      4           70178.00   2.00       Dec 2 16:08
code      matilde    50083     452         68          226.00     34.00      Dec 2 20:37
code      matilde    12390     21          8           10.50      4.00       Dec 2 11:48
chrome    matilde    69933     1529        0           764.50     0          Dec 2 21:13
chrome    matilde    6768      104         0           52.00      0          Dec 2 11:17
chrome    matilde    5016      141         66125       70.50      33062.50   Dec 2 11:16
chrome    matilde    4967      46          1102        23.00      551.00     Dec 2 11:16
chrome    matilde    49143     1530        0           765.00     0          Dec 2 20:23
chrome    matilde    268503    23          23          11.50      11.50      Dec 2 23:00
```

## Execuções utilizando vários comandos e argumentos de entrada

```
matilde@matilde-XPS-15-9520:~/S0/projeto$ ./rwstat.sh -m 3000 -w 1
```

COMM	USER	PID	READB	WRITEB	RATER	RATEW	DATE
rwstat.sh	matilde	362913	72644586	404966	72644586.00	404966.00	Dec 2 23:39
gnome-control-c	matilde	32852	3200	9560	3200.00	9560.00	Dec 2 17:39
Xwayland	matilde	3937	0	3072	0	3072.00	Dec 2 11:16
pulseaudio	matilde	3202	2691	2691	2691.00	2691.00	Dec 2 11:16
solaar	matilde	3611	5463	1760	5463.00	1760.00	Dec 2 11:16
Discord	matilde	27560	101392	930	101392.00	930.00	Dec 2 16:08
gnome-shell	matilde	3408	264	280	264.00	280.00	Dec 2 11:16
Discord	matilde	27488	130	130	130.00	130.00	Dec 2 16:08
gsd-sharing	matilde	3580	24	40	24.00	40.00	Dec 2 11:16
code	matilde	12466	0	31	0	31.00	Dec 2 11:48
TabNine	matilde	50152	0	26	0	26.00	Dec 2 20:37
code	matilde	50083	8	21	8.00	21.00	Dec 2 20:37
chrome	matilde	4967	16	16	16.00	16.00	Dec 2 11:16
code	matilde	12390	235	8	235.00	8.00	Dec 2 11:48
code	matilde	12447	8	8	8.00	8.00	Dec 2 11:48
Discord	matilde	27359	93574	2	93574.00	2.00	Dec 2 16:08
chrome	matilde	5016	110	0	110.00	0	Dec 2 11:16
pipewire-media-	matilde	3201	992	0	992.00	0	Dec 2 11:16

```
matilde@matilde-XPS-15-9520:~/S0/projeto$ ./rwstat.sh -c "c.*" -M 4000 -r 1
```

COMM	USER	PID	READB	WRITEB	RATER	RATEW	DATE
gnome-control-c	matilde	32852	3024	9072	3024.00	9072.00	Dec 2 17:39
Discord	matilde	27560	590	971	590.00	971.00	Dec 2 16:08
Discord	matilde	27488	135	135	135.00	135.00	Dec 2 16:08
Discord	matilde	27359	140365	3	140365.00	3.00	Dec 2 16:08
code	matilde	50083	8	21	8.00	21.00	Dec 2 20:37
code	matilde	12466	0	31	0	31.00	Dec 2 11:48
code	matilde	12447	8	8	8.00	8.00	Dec 2 11:48
chrome	matilde	6768	384	24	384.00	24.00	Dec 2 11:17
chrome	matilde	5016	11	11	11.00	11.00	Dec 2 11:16
chrome	matilde	4967	22359	70623	22359.00	70623.00	Dec 2 11:16
chrome	matilde	268503	50	45	50.00	45.00	Dec 2 23:00

```
matilde@matilde-XPS-15-9520:~/S0/projeto$ ./rwstat.sh -c "c.*" -M 4000000 -r 1
```

COMM	USER	PID	READB	WRITEB	RATER	RATEW	DATE
gnome-control-c	matilde	32852	3024	9072	3024.00	9072.00	Dec 2 17:39
Discord	matilde	27560	590	971	590.00	971.00	Dec 2 16:08
Discord	matilde	27488	135	135	135.00	135.00	Dec 2 16:08
Discord	matilde	27359	140365	3	140365.00	3.00	Dec 2 16:08
code	matilde	50083	8	21	8.00	21.00	Dec 2 20:37
code	matilde	12466	0	31	0	31.00	Dec 2 11:48
code	matilde	12447	8	8	8.00	8.00	Dec 2 11:48
chrome	matilde	6768	384	24	384.00	24.00	Dec 2 11:17
chrome	matilde	5016	11	11	11.00	11.00	Dec 2 11:16
chrome	matilde	4967	22359	70623	22359.00	70623.00	Dec 2 11:16
chrome	matilde	268503	50	45	50.00	45.00	Dec 2 23:00

```
matilde@matilde-XPS-15-9520:~/S0/projeto$ ./rwstat.sh -u matilde -e "Dec 02 13:00" -w 1
```

COMM	USER	PID	READB	WRITEB	RATER	RATEW	DATE
Xwayland	matilde	3937	0	3276	0	3276.00	Dec 2 11:16
pulseaudio	matilde	3202	2648	2648	2648.00	2648.00	Dec 2 11:16
solaar	matilde	3611	5415	1740	5415.00	1740.00	Dec 2 11:16
gnome-shell	matilde	3408	240	240	240.00	240.00	Dec 2 11:16
TabNine-deep-lo	matilde	7679	0	39	0	39.00	Dec 2 11:18
chrome	matilde	4967	15	15	15.00	15.00	Dec 2 11:16
code	matilde	12390	21	8	21.00	8.00	Dec 2 11:48
pipewire-media-	matilde	3201	992	0	992.00	0	Dec 2 11:16

```
matilde@matilde-XPS-15-9520:~/S0/projeto$ ./rwstat.sh -u matilde -e "Dec 02 13:00" -p 3 -w 1
```

COMM	USER	PID	READB	WRITEB	RATER	RATEW	DATE
chrome	matilde	5016	151	82539	151.00	82539.00	Dec 2 11:16
chrome	matilde	4967	17889	57045	17889.00	57045.00	Dec 2 11:16
Xwayland	matilde	3937	0	3212	0	3212.00	Dec 2 11:16

```
matilde@matilde-XPS-15-9520:~/S0/projeto$ ./rwstat.sh -u matilde -s "Dec 02 11:00" -e "Dec 02 13:00" -p 3 -w 1
```

COMM	USER	PID	READB	WRITEB	RATER	RATEW	DATE
chrome	matilde	5016	110	82524	110.00	82524.00	Dec 2 11:16
chrome	matilde	4967	18	12318	18.00	12318.00	Dec 2 11:16
pulseaudio	matilde	3202	2564	2564	2564.00	2564.00	Dec 2 11:16

```
matilde@matilde-XPS-15-9520:~/S0/projeto$ ./rwstat.sh -m 3 -M 30000 -c "d.*" -p 3 -w 1
```

COMM	USER	PID	READB	WRITEB	RATER	RATEW	DATE
Xwayland	matilde	3937	0	42648	0	42648.00	Dec 2 11:16
pulseaudio	matilde	3202	2632	2632	2632.00	2632.00	Dec 2 11:16
Discord	matilde	27560	454	835	454.00	835.00	Dec 2 16:08

## Conclusão

A resolução deste trabalho foi muito educativa ao aprofundar os nossos conhecimentos sobre bash, quer a partir do material teórico e prático das aulas de Sistemas Operativos, quer a partir da pesquisa impulsionada pelos constantes obstáculos que surgiram na realização deste trabalho.

Este trabalho foi um sucesso, pois os resultados obtidos são iguais aos esperados, tendo a realização deste nos motivado para expandirmos os nossos horizontes sobre a linguagem bash.