

Ferramenta de criação/atualização de cópias de segurança em bash

Índice

Introdução.....	3
Metodologia.....	3
Backup_files.sh.....	4
1)Execução do programa.....	4
2)Tratamento de opções.....	4
3)Integridade de argumentos.....	4
4)Função backup_files().....	5
Backup.sh.....	6
1)Novas Opções.....	6
2)Função read_exclusion_list().....	6
3)Função Backup_gen().....	7
4)Função execute().....	7
Backup_summary.sh.....	8
Backup_check.sh.....	9
Testes de validação.....	10
Conclusão.....	11
Webgrafia.....	11

Introdução

O objetivo deste projeto é desenvolver um script em shell bash que possibilite gerar uma cópia backup de um dado diretório.

Neste relatório abordamos o processo que realizámos para chegar à resolução do problema proposto, fazendo uso de amostras do código-fonte.

De maneira a criar um paralelismo com a metodologia descrita no guião que nos foi fornecido, separamos a análise do código em várias partes, correspondendo às fases de desenvolvimento do script.

Para correr o programa é necessário passar como argumentos o caminho do diretório que pretendemos salvar para além do caminho de destino onde queremos o nosso backup seja criado.

Em complemento ao seu propósito principal e como pedido no guião, a partir da 3ª versão, no fim de execução o programa apresenta um sumário que inclui o número de warnings, ficheiros atualizados, ficheiros copiados, ficheiros apagados e o número de erros, estatísticas que quando relacionadas nos ajudam a avaliar a performance do mesmo.

Metodologia

O nosso projeto foi dividido em 4 fases de desenvolvimento, em cada uma adicionamos uma nova funcionalidade.

A primeira faz backup restritamente da diretoria raiz em relação ao caminho passado, ignorando ficheiros contidos em quaisquer diretorias descendentes.

A segunda passa a iterar recursivamente sobre todas as diretorias-filhas até não existirem mais diretorias por onde iterar.

A terceira imprime o sumário na consola com informações estatísticas em relação à execução do programa.

A função final, verifica se houve mudança nos ficheiros da diretoria original (criação, remoção ou modificação de ficheiros) e imprime na consola as mesmas.

Este metodo de elaboração do script permite-nos fragmentar o problema principal em problemas mais simples, de maneira a podermos raciocinar soluções de forma mais simplista e eficiente.

Como a versão seguinte é uma versão aprimorada da anterior neste relatório se uma versão usa um segmento de código que se manteve constante em relação à versão anterior decidimos deixá-lo omissa, partindo do princípio que seria desnecessário explicar o mesmo segmento novamente.

Backup_files

• Execução do programa

Para executar o programa passamos como primeiro argumento o caminho para o diretório que pretendemos salvar e de seguinte o caminho destino, caso queiramos correr o script em modo checking (Não realiza os comandos, apenas faz impressões) usamos a opção “-c”, como exemplificado na imagem:

```
./Backup_files.sh -c /home/belito/Documents/GitHub/Backup_S0/Projeto/Receitas_Original /home/belito/Documents/GitHub/Backup_S0
```

• Tratamento de opções

No início de execução a variável `$check_flag` que dita se o programa será corrido no modo normal ou checking e está definida pra false como default.

```
check_flag=false
while getopts ":c" opt; do
  case $opt in
    c) check_flag=true
       ;;
    \?) echo "Invalid option: -$OPTARG" >&2
        exit 1
        ;;
    esac
done
shift $((OPTIND-1))
```

`getopts` é usado para processar as opções da linha de comandos, “:c” `opt` define -c como uma opção possível.

Case começa um switch case relativamente ao valor de `$opt` caso seja “c” o programa entra em modo checking, se for qualquer outro o programa retorna um erro.

`shift $((OPTIND-1))` remove dos argumentos as opções já processadas de modo a que os argumentos `$1` e `$2` sejam os caminhos.

A função principal, `main()`, é chamada com estes dois argumentos, que são guardados nas variáveis `$starting_dir` e `$end_dir` respetivamente.

• Integridade de argumentos

Primeiramente verificamos a integridade dos caminhos passados na linha de comandos através da função:

```
function check_dir_integ(){
  if [ -z "$1" ] || [ ! -d "$1" ]; then
    echo "Error: Source directory '$1' is not a valid path."
    exit 1
  fi

  if [ -z "$2" ] || [ ! -d "$2" ]; then
    echo "Error: Destination directory '$2' is not a valid path."
    exit 1
  fi
}
```

`[-z "$arg"]` verifica se o diretório está vazio (“-z” retorna true se “\$1” tiver length 0) e `[! -d “arg”]` verifica se o diretório é um diretório válido (se o caminho não aponta pra outro tipo de documento).

Para conseguirmos estabelecer um padrão que funcione independentemente do tipo do caminho dos argumentos (relativo ou absoluto) fazemos o seguinte:

```
starting_dir=$1
if [[ "$starting_dir" != /* ]]; then
  starting_dir=$(realpath "$1")
fi
end_dir=$2
if [[ "$end_dir" != /* ]]; then
  end_dir=$(realpath "$2")
fi
```

As condições dos ifs verificam se os caminhos não começam com “/” o que indica se são caminhos relativos, caso o sejam o comando `realpath` converte-o para o caminho absoluto correspondente.

De seguida anexamos ao caminho da diretoria destino o nome do diretório original + `_backup`, para podermos criar a pasta de backup caso ela não exista (o comando `basename` retorna o nome de o ficheiro ou diretório dado o caminho do mesmo).

```
path_diretoria_destino="$end_dir/${basename "$starting_dir"}_backup"
```

• Função Backup_files()

```
if $check_flag; then
  if [ ! -e "$path_diretoria_destino" ]; then
    echo "mkdir" "$path_diretoria_destino"
    Backup_files_c "$starting_dir" "$path_diretoria_destino"
  else
    Backup_files_c "$starting_dir" "$path_diretoria_destino"
  fi
else
  if [ ! -e "$path_diretoria_destino" ]; then
    echo "$path_diretoria_destino does not exist."
    mkdir "$path_diretoria_destino"
    Backup_files "$starting_dir" "$path_diretoria_destino"
    echo "$path_diretoria_destino has been created"
  else
    echo "$path_diretoria_destino already exists."
    Backup_files "$starting_dir" "$path_diretoria_destino"
  fi
fi
```

Este segmento verifica a existência da diretoria destino, se não existir cria-a e chama a função `Backup_files()`, se existir apenas chama a função. Caso o modo checking esteja ativado não executa as funções, apenas as imprime no terminal e chama uma versão alternativa de `Backup_files` a `Backup_files_c()` semelhante mas que também não executa nenhum dos comandos (na versão seguinte implementámos algo diferente de maneira a eliminar esta redundância).

A função `Backup_files()` define as variáveis anteriores dentro do seu domínio e inicia um for loop com a seguinte expressão de iteração:

```
for file in "$diretoria_atual"/*; do
```

Aqui a variável `$diretoria_atual` tem novamente o caminho da diretoria a copiar (este nome já ficou definido a pensar na versão posterior), `*` expande para uma lista com todos os ficheiros presentes na diretoria anterior.

Logo esta expressão traduz-se em, para todos os ficheiros na diretoria atual faz o seguinte:

```
if [ -f "$file" ]; then
  if [ ! -e "$path_diretoria_destino/${basename $file}" ]; then
    echo "cp -a" "$file" "$path_diretoria_destino"

  elif [ $file -nt $path_diretoria_destino ]; then
    echo "File ${basename $file} has been updated. Backing up now."
    echo "cp -a" "$file" "$path_diretoria_destino"
    cp -a "$file" "$path_diretoria_destino"
  fi
fi
```

A condição `[-f "$file"]` verifica se o ficheiro da iteração atual é um ficheiro standard (se não é um diretório).

A condição `[! -e "$path_diretoria_destino/${basename $file}"]` anexa ao caminho destino o nome do ficheiro a copiar e verifica se o mesmo existe, se não existir copia-o para a diretoria destino.

Se já existir `[$file -nt $path_diretoria_destino]` compara as datas de modificação dos dois ficheiros, se o ficheiro original for mais recente que o ficheiro de backup, significa que existe uma versão mais recente do ficheiro logo copiamos esta versão.

Backup

• Novas opções

Este script implementa algumas novas funcionalidades que dependem do uso de opções na altura de executar o mesmo. Portanto a secção em que as processamos foi atualizada da seguinte forma:

```
while getopts ":cb:r:" opt; do
  case $opt in
    c)
      flag_c=true
      ;;
    b)
      flag_b=true
      file_txt="$OPTARG"
      if [ ! -f "$file_txt" ]; then
        echo "Error: Backup file '$file_txt' not found."
        exit 1
      fi
      [[ "$file_txt" != /* ]] && file_txt=$(realpath "$file_txt")
      ;;
    r)
      flag_r=true
      expression="$OPTARG"
      ;;
    \?)
      echo "Invalid option: -$OPTARG" >&2
      exit 1
      ;;
    *)
      ;;
  esac
done
```

Para além da flag já existente "-c", agora temos "-b" e "-r", que indicam, respetivamente, um ficheiro com nomes de documentos que o script deve ignorar e que apenas devem ser copiados os ficheiros que verificam uma dada expressão regular, para além de flags que indicam que opções foram passadas.

Este segmento verifica se o numero de argumentos passados "\$#" é diferente de 2, que indica uma tentativa inadequada de execução do programa, e termina o programa imprimindo o formato correto de execução:

```
if [ "$#" -ne 2 ]; then
  echo "Usage: $0 [-c] [-b backup_file] [-r] source_path dest_path"
  exit 1
fi
```

```
if $flag_b; then
  read_exclusion_list "$file_txt"
fi
```

Na função main acrescentamos esta condição que chama a função `read_exclusion_list()` conforme a existência da opção -b.

• Função read_exclusion_list()

A função `read_exclusion_list()` é chamada com o ficheiro com os nomes para exclusão como argumento, começa então por criar uma lista para armazenar o conteúdo do ficheiro.

```
function read_exclusion_list() {
  local exclusion_file="$1"
  exclusion_list=()
  while IFS= read -r line; do
    [[ "$line" != /* ]] && line="$starting_dir/$line"
    exclusion_list+=("$line")
  done < "$exclusion_file"
}
```

Na inicialização do while loop, mudamos temporariamente o IFS para um valor nulo de modo a preservar a integridade de cada linha lida do ficheiro. `$line` tem o valor da linha lida

na correspondente iteração, verifica-se se o ficheiro é um caminho relativo e caso seja adicionamos o prefixo necessário e acrescentamo-lo á lista.

- **Função Backup_gen()**

Chamamos a função `backup_gen()` e primeiro verificamos se o ficheiro pretence á lista de ficheiros excluidos através da função `is_excluded()`, se for passa-se à proxima iteração.

```
function is_excluded() {
    local path="$1"
    for excluded in "${exclusion_list[@]}; do
        [[ "$path" == "$excluded" ]] && return 0
    done
    return 1
}
```

De seguida, definimos algumas variaveis necessarias para iterarmos sobre os ficheiros:

```
local diretorio_atual="$1"
local path_diretorio_destino="$2"

for file in "$diretorio_atual"/*; do

    if is_excluded "$file"; then
        echo "Skipping excluded file or directory: $file"
        continue
    fi

    path_backup_file="$path_diretorio_destino/${file#starting_dir/}"
    path_backup_dir="$path_diretorio_destino/${diretorio_atual#starting_dir/}"
    path_original_dir="$starting_dir/${diretorio_atual#starting_dir/}"

    if [ "$diretorio_atual" == "$starting_dir" ];then
        path_backup_dir="$path_diretorio_destino"
        path_original_dir="$starting_dir"
    fi
```

Verifica-se existe a diretorio do backup referente a iteração atual, caso não exista é criada.

```
if [ ! -e "$path_backup_file" ]; then
    [ ! -e "$backup_dir" ] && execute mkdir -p "$backup_dir"

    if [ -f "$file" ]; then
        if [[ ! "$file" =~ $expression ]] && [[ $flag_r ]]; then
            echo "Skipping $file: doesnt match the provided expression. "
            continue
        elif [ ! -e "$path_backup_file" ]; then
            execute cp -a "$file" "$path_backup_file"
        fi

    elif [ -d "$file" ]; then
        backup_gen "$file" "$path_diretorio_destino"
    fi
```

Se `$file` for um ficheiro normal, na hipotese de não ser congruente com a expressão regular definida e a opção `"-r"` tiver sido usada na execução, passa-se á proxima iteração.

De outro modo se o ficheiro não existir é copiado.

- **Função Execute()**

Se for uma diretorio chama-se novamente a função `backup_gen()` com a mesma como argumento.

```
execute() {
    if [ "$flag_c" = "false" ];then
        eval "${@}" || return;
    fi
    echo "${@}"
}
```

Neste trecho e no resto do codigo dispomos da função `execute()` sempre que copiamos/removemos um ficheiro ou criamos uma diretorio. Esta função visa substituir a redundancia de ter uma função separada caso o script seja corrido em modo check, utilizando o comando `eval`. `"@"` recebe os argumentos

da função `execute()`, eval recebe esses argumentos como string, caso esta corresponda a um comando bash válido, executa-a como se diretamente na shell. Na eventualidade de eval "{@}" falhar, a função retorna logo em seguida sem imprimir o comando executado.

```
if [ ! -e $file ]; then
    execute rm $path_backup_file;
else
```

Tudo isto se verifica se o `$file` da iteração não existir no backup.

Caso contrário verificamos se o ficheiro não foi atualizado através da condição `-nt` que já usamos na versão anterior.

No fim de iterarmos sobre a diretoria atual, comparamos o total de ficheiros na pasta com o total de ficheiros na pasta backup correspondente.

```
count_backup=$(ls -l "$path_backup_dir" | wc -l)
count_original=$(ls -l "$path_original_dir" | wc -l)

if [ "$count_backup" -gt "$count_original" ]; then
    for backupfile in "$path_backup_dir"/*; do
        filename=$(basename "$backupfile")

        if [ ! -e "$path_original_dir/$filename" ]; then
            if [ -f "$backupfile" ]; then
                execute rm "$backupfile"
            elif [ -d "$backupfile" ]; then
                execute rm -r "$backupfile"
            fi
        fi
    done
fi
```

Se o número de ficheiros for maior na pasta backup existe algum ficheiro na mesma que já não está presente no diretório de trabalho. `$(ls -l path/ | wc -l)` devolve o número de ficheiros em `"path"`.

Sendo assim iteramos sobre o diretório atual de backup e verificamos a existência de cada ficheiro na diretoria de trabalho.

Backup_Summary

Nesta versão implementamos a contagem de algumas estatísticas, nomeadamente número de erros, warnings, ficheiros copiados e ficheiros atualizados. Estes dados são importantes tanto para avaliar a performance do nosso programa como para confirmar a correta execução do mesmo, todos eles são impressos no terminal da iteração de cada sub diretório.

```
local copied=0
local updated=0
local errors=0
local deleted=0
local size_deleted=0
local size_copied=0
```

No início da função `backup_gen()` definimos as variáveis de contagem como locais para que a contagem de uma subdiretória não afete as contagens de qualquer outra.

Incrementamos ficheiros copiados:

```
elif [ ! -e "$path_backup_file" ]; then
    execute cp -a "$file" "$path_backup_file"
    if [ $? -eq 0 ]; then
        ((copied++))
        size_copied=$((size_copied + $(stat --format="%s" "$file")))
    fi
fi
```

Quando verificamos que o ficheiro não existe e `execute()` realiza o cp com sucesso.

Incrementamos os warnings:

```
if [ ! -e $file ]; then
    (( errors++ ))
    execute rm $path_backup_file;
else
```

Quando encontramos um ficheiro no backup que não está presente na diretoria original.


```
execute() {
    if [ "$flag_c" = "false" ];then
        eval "${@}" || {(errors++);return 1;}
    fi

    echo "${@}"
}
```

Quando o comando **eval** “{@}” não retorna executa com sucesso.

Incrementamos os ficheiros atualizados:

```
elif [ $path_backup_file -ot $file ];then
    execute "cp -a" "$file" "$path_backup_file"
    [ $? -eq 0 ] && ((updated++))
fi
```

Quando detetamos uma versão mais recente do ficheiro.

Incrementamos ficheiros apagados:

```
if [ -f "$backupfile" ]; then
    aux_cum=$(stat --format=%s "$backupfile")
    aux_count=1
    execute rm "$backupfile"

elif [ -d "$backupfile" ]; then
    aux_cum=$(du -sb "$backupfile" | cut -f1)
    aux_count=$(find "$backupfile" -type f | wc -l)
    execute rm -r "$backupfile"
fi

if [ $? -eq 0 ]; then
    size_deleted=$((size_deleted + aux_cum))
    deleted=$((deleted + aux_count))
fi
```

Quando verificamos que um ficheiro se encontra na diretoria backup, mas não na original.

Se for um ficheiro normal aumentamos normalmente, se for uma diretoria calculamos o número e tamanho cumulativo dos ficheiros removidos e adicionamos.

Em ambos os casos só incrementamos após verificar que **execute()** teve sucesso no comando remove.

No final de cada iteração ou seja por cada subdiretoria imprimimos no terminal as contagens correspondentes.

```
echo "While backing $diretoria_atual: $errors warnings, $updated updated, $copied copied (${size_copied}B) and $deleted deleted (${size_deleted}B)."
```

Backup_check

Este script usa a função **md5sum** para comparar o conteúdo do ficheiro presente no backup com o correspondente da diretoria original, de forma a verificarmos se houve alguma modificação.

```
function check_dir_integ(){
    if [ -z "$1" ] || [ ! -d "$1" ]; then
        echo "Error: Source directory '$1' is not a valid path."
        exit 1
    fi

    if [ -z "$2" ] || [ ! -d "$2" ]; then
        echo "Error: Destination directory '$2' is not a valid path."
        exit 1
    fi
}
```

Os argumentos na execução do programa são os mesmo, e é verificada a sua validade com a mesma função dos programas anteriores.

Usamos também a recursiva anterior para iterar sob todos os ficheiro de ambos os diretorios. Caso o ficheiro da atual iteração seja um ficheiro normal (não diretorio), usamos a função `md5sum` para obter o hashcode correspondentes aos ficheiros da dire original e backup, que guardamos nas variaveis `md5_source` e `md5_backup`

```
md5_source=$(md5sum "$file" | awk '{ print $1 }')
md5_backup=$(md5sum "$path_backup_file" | awk '{ print $1 }')
```

Como `md5sum` para além do hashcode retorna o nome do ficheiro sob o qual aplicou a função hash, usamos `awk '{print $1}'` para ignorar essa parte to retorno

```
if [ "$md5_source" != "$md5_backup" ]; then
    echo "$file $path_backup_file differ."
fi
else dir_checker "$file" "$path_diretoria_destino"
```

Comparamos os hashcodes resultantes dos dois ficheiros, o facto de não serem iguais indica-nos que um dos ficheiros foi modificados logo imprimimos um aviso no terminal com essa informação.

Testes

A realização de testes é fundamental para garantir que o trabalho desenvolvido cumpre de maneira satisfatoria com as normas indicadas e que irá funcionar corretamente frente às mais diversas condições de execução.

Os testes que efetuamos ao nosso código visaram tanto coadjuvar no seu desenvolvimento aquando de novas implementações, como testar a robustez e confiabilidade do mesmo, testando desde as situações de utilização comuns e expectáveis a argumentos de entrada inválidos.

Quando realizavamos alguma implementação obviamente manipulavamos as diretorias e os ficheiros de maneira a provocar erros que normalmente não deveriam acontecer, mudar as permissões de um ficheiro na diretoria original por exemplo, este tipo de testes não vai ser citado, pois achamos que esteja implicito que se existe codigo implementado para essas situações, já realizamos os devidos testes.

- **Validação de argumentos**

Descrição	Entrada de Teste	Resultado
Caminhos Completos	/home/belito/Documents/GitHub/Backup_SO/Projeto/Receitas_Original /home/belito/Documents/GitHub/Backup_SO	Execução Normal
Caminhos relativos	GitHub/Backup_SO/Projeto/Receitas_Original GitHub/Backup_SO	Exeução Normal
Argumentos inválidos	Skibidbapum123 123423567	Erro de controlo (proposital)
Número Inválido de Argumentos	1 arg - /home/belito/Documents/GitHub/Backup_SO/Projeto/Receitas_Original	Erro de controlo
Inserção incorreta de opções	/home/belito/Documents/GitHub/Backup_SO/Projeto/ Receitas_Original /home/belito/Documents/GitHub/Backup_SO -c	Erro de controlo
Caminho original == Backup	/home/belito/Documents/GitHub/Backup_SO/Projeto/Receitas_Original /home/belito/Documents/GitHub/Backup_SO/Projeto/Receitas_Original	Má execução (corrigido)
Diretorios sem permissões rwx	chmod -rwx Backup_SO /home/belito/Documents/GitHub/Backup_SO/Projeto/Receitas_Original /home/belito/Documents/GitHub/Backup_SO	Má execução (corrigido)

Regex inválidos	-r "([a-z" -r "*" +[" -r "(a*\)"	Execução Normal
Ficheiro de exclusão inexistente	-b skibidibapum /home/belito/Documents/GitHub/Backup_SO/Projeto/Receitas_Original /home/belito/Documents/GitHub/Backup_SO	Erro de controlo

Testamos também o script a partir do programa test_a1 que nos foi fornecido, onde inicialmente obtivemos 15 pontos, verificamos que isto se devia a dois problemas.

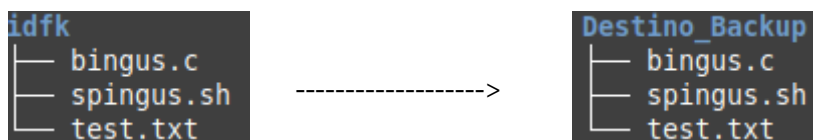
Primeiramente na nossa implementação tomamos a liberdade de através da diretoria original, criar a backup com o sufixo “_backup”, o ficheiro de teste não contava com isto logo os ficheiros não eram copiados corretamente, por bem retiramos esse segmento do nosso código e obtivemos 40 pontos.

Verificamos também que o nosso output não correspondia ao estabelecido pelo programa, em grande parte porque usávamos os caminhos absolutos em vez de caminhos relativos. Mudar isto obrigou-nos a reestruturar o uso da função execute(), que não satisfazia esta exigência. Formatos os outputs e passamos a obter os 100 pontos desejados.

- Testes de uso

-Backup_files.sh

-Teste normal :



```
sleepyy@sleepyy-L0Q-15APH8:~/Backup_SO/Projeto$ ./backup_files.sh idfk Destino_Backup
cp -a idfk/bingus.c Destino_Backup/bingus.c
cp -a idfk/spingus.sh Destino_Backup/spingus.sh
cp -a idfk/test.txt Destino_Backup/test.txt
```

-Teste a fazer uso de flag -c:

```
sleepyy@sleepyy-L0Q-15APH8:~/Backup_SO/Projeto$ ./backup_files.sh -c idfk Destino_Backup
cp -a idfk/bingus.c Destino_Backup/bingus.c
cp -a idfk/spingus.sh Destino_Backup/spingus.sh
cp -a idfk/test.txt Destino_Backup/test.txt
```

-Teste de atualização de ficheiros depois de já existir copia:

```
sleepyy@sleepyy-L0Q-15APH8:~/Backup_SO/Projeto$ ./backup_files.sh idfk Destino_Backup
idfk/test.txt has been updated.
cp -a idfk/test.txt Destino_Backup/test.txt
```

-Backup.sh e -Backup_summary.sh

Apesar de testarmos para ambos os scripts, como o código base é fundamentalmente igual, apenas muda o output, decidimos apresentar apenas os tests para Backup_summary.sh.

Sem uso de opções

Diretoria de trabalho - /home/belito/Documents/GitHub/Backup_SO/Projeto/Receitas_Original

Diretoria de Backup - /home/belito/Documents/GitHub/Backup_SO/backup
(Inexistente)

Output

```
mkdir backup/_backup
cp -a Receitas_Original/_backup/outro_ficheiro.txt backup/_backup/outro_ficheiro.txt
While backuping Receitas_Original/_backup: 0 Errors; 0 Warnings; 0 Updated; 1 Copied (12B); 0 Deleted (0B)
mkdir backup/Bolos
cp -a Receitas_Original/Bolos/Bolo_Chocolate.txt backup/Bolos/Bolo_Chocolate.txt
cp -a Receitas_Original/Bolos/Bolo_Marmore.txt backup/Bolos/Bolo_Marmore.txt
cp -a Receitas_Original/Bolos/ficheiro_sem_permissoes.txt backup/Bolos/ficheiro_sem_permissoes.txt
While backuping Receitas_Original/Bolos: 0 Errors; 0 Warnings; 0 Updated; 3 Copied (4361B); 0 Deleted (0B)
mkdir backup/Carne
cp -a Receitas_Original/Carne/Cabidela.txt backup/Carne/Cabidela.txt
cp -a Receitas_Original/Carne/Chanfana.txt backup/Carne/Chanfana.txt
While backuping Receitas_Original/Carne: 0 Errors; 0 Warnings; 0 Updated; 2 Copied (2677B); 0 Deleted (0B)
cp -a Receitas_Original/dsdasdasda backup/dsdasdasda
mkdir backup/Massas
mkdir backup/Massas/Italiana
cp -a Receitas_Original/Massas/Italiana/Bolenhesa.txt backup/Massas/Italiana/Bolenhesa.txt
cp -a Receitas_Original/Massas/Italiana/ficheiro_para_eliminar.txt backup/Massas/Italiana/ficheiro_para_eliminar.txt
cp -a Receitas_Original/Massas/Italiana/hbjb backup/Massas/Italiana/hbjb
cp -a Receitas_Original/Massas/Italiana/Lasanha.txt backup/Massas/Italiana/Lasanha.txt
While backuping Receitas_Original/Massas/Italiana: 0 Errors; 0 Warnings; 0 Updated; 4 Copied (1762B); 0 Deleted (0B)
cp -a Receitas_Original/Massas/Massas_muito_bom.txt backup/Massas/Massas_muito_bom.txt
While backuping Receitas_Original/Massas: 0 Errors; 0 Warnings; 0 Updated; 1 Copied (252B); 0 Deleted (0B)
mkdir backup/src
mkdir backup/src/aaa
cp -a Receitas_Original/src/aaa/prog2.c backup/src/aaa/prog2.c
While backuping Receitas_Original/src/aaa: 0 Errors; 0 Warnings; 0 Updated; 1 Copied (8282B); 0 Deleted (0B)
cp -a Receitas_Original/src/output.txt backup/src/output.txt
cp -a Receitas_Original/src/prog1.c backup/src/prog1.c
cp -a Receitas_Original/src/start.sh backup/src/start.sh
While backuping Receitas_Original/src: 0 Errors; 0 Warnings; 0 Updated; 3 Copied (16596B); 0 Deleted (0B)
cp -a Receitas_Original/ssdasds backup/ssdasds
While backuping Receitas_Original: 0 Errors; 0 Warnings; 0 Updated; 2 Copied (27B); 0 Deleted (0B)
```

Diretoria Original

```
Receitas_Original
├── backup
│   └── outro_ficheiro.txt
├── Bolos
│   ├── Bolo_Chocolate.txt
│   ├── Bolo_Marmore.txt
│   └── ficheiro_sem_permissoes.txt
├── Carne
│   ├── Cabidela.txt
│   └── Chanfana.txt
├── dsdasdasda
├── Massas
│   └── Italiana
│       ├── Bolenhesa.txt
│       ├── ficheiro_para_eliminar.txt
│       ├── hbjb
│       └── Lasanha.txt
│       └── Massas_muito_bom.txt
├── src
│   ├── aaa
│   │   ├── prog2.c
│   │   ├── output.txt
│   │   ├── prog1.c
│   │   └── start.sh
│   └── ssdasds
```

Diretoria Original

```
backup
├── backup
│   └── outro_ficheiro.txt
├── Bolos
│   ├── Bolo_Chocolate.txt
│   ├── Bolo_Marmore.txt
│   └── ficheiro_sem_permissoes.txt
├── Carne
│   ├── Cabidela.txt
│   └── Chanfana.txt
├── dsdasdasda
├── Massas
│   └── Italiana
│       ├── Bolenhesa.txt
│       ├── ficheiro_para_eliminar.txt
│       ├── hbjb
│       └── Lasanha.txt
│       └── Massas_muito_bom.txt
├── src
│   ├── aaa
│   │   ├── prog2.c
│   │   ├── output.txt
│   │   ├── prog1.c
│   │   └── start.sh
│   └── ssdasds
```

Opções - “-c”

Diretoria de trabalho - /home/belito/Documents/GitHub/Backup_SO/Projeto/Receitas_Original

Diretoria de Backup - /home/belito/Documents/GitHub/Backup_SO/backupup
(Inexistente)

Output

```
mkdir backupup/ backup
cp -a Receitas_Original/_backup/outro_ficheiro.txt backupup/_backup/outro_ficheiro.txt
While backuping Receitas_Original/_backup: 0 Errors; 0 Warnings; 0 Updated; 1 Copied (12B); 0 Deleted (0B)
mkdir backupup/Bolos
cp -a Receitas_Original/Bolos/Bolo_Chocolate.txt backupup/Bolos/Bolo_Chocolate.txt
cp -a Receitas_Original/Bolos/Bolo_Marmore.txt backupup/Bolos/Bolo_Marmore.txt
cp -a Receitas_Original/Bolos/ficheiro sem permissoes.txt backupup/Bolos/ficheiro sem permissoes.txt
While backuping Receitas_Original/Bolos: 0 Errors; 0 Warnings; 0 Updated; 3 Copied (4361B); 0 Deleted (0B)
mkdir backupup/Carne
cp -a Receitas_Original/Carne/Cabidela.txt backupup/Carne/Cabidela.txt
cp -a Receitas_Original/Carne/Chanfana.txt backupup/Carne/Chanfana.txt
While backuping Receitas_Original/Carne: 0 Errors; 0 Warnings; 0 Updated; 2 Copied (2677B); 0 Deleted (0B)
cp -a Receitas_Original/dsdasdasda backupup/dsdasdasda
mkdir backupup/Massas
mkdir backupup/Massas/Italiana
cp -a Receitas_Original/Massas/Italiana/Bolenhesa.txt backupup/Massas/Italiana/Bolenhesa.txt
cp -a Receitas_Original/Massas/Italiana/ficheiro para eliminar.txt backupup/Massas/Italiana/ficheiro para eliminar.txt
cp -a Receitas_Original/Massas/Italiana/hbjb backupup/Massas/Italiana/hbjb
cp -a Receitas_Original/Massas/Italiana/Lasanha.txt backupup/Massas/Italiana/Lasanha.txt
While backuping Receitas_Original/Massas/Italiana: 0 Errors; 0 Warnings; 0 Updated; 4 Copied (1762B); 0 Deleted (0B)
cp -a Receitas_Original/Massas/Massas_muito_bom.txt backupup/Massas/Massas_muito_bom.txt
While backuping Receitas_Original/Massas: 0 Errors; 0 Warnings; 0 Updated; 1 Copied (252B); 0 Deleted (0B)
mkdir backupup/src
mkdir backupup/src/aaa
cp -a Receitas_Original/src/aaa/prog2.c backupup/src/aaa/prog2.c
While backuping Receitas_Original/src/aaa: 0 Errors; 0 Warnings; 0 Updated; 1 Copied (8282B); 0 Deleted (0B)
cp -a Receitas_Original/src/output.txt backupup/src/output.txt
cp -a Receitas_Original/src/progl.c backupup/src/progl.c
cp -a Receitas_Original/src/start.sh backupup/src/start.sh
While backuping Receitas_Original/src: 0 Errors; 0 Warnings; 0 Updated; 3 Copied (16596B); 0 Deleted (0B)
cp -a Receitas_Original/ssdasds backupup/ssdasds
While backuping Receitas_Original: 0 Errors; 0 Warnings; 0 Updated; 2 Copied (27B); 0 Deleted (0B)
```

Opções - “-b ex.txt”

Diretoria de trabalho - /home/belito/Documents/GitHub/Backup_SO/Projeto/Receitas_Original

Diretoria de Backup - /home/belito/Documents/GitHub/Backup_SO/backupup
(Inexistente)

ex.txt

```
ex.txt X
/home/belito/Documents/GitHub/Backup_SO/Projeto/Receitas_Original
Massas_muito_bom.txt
Bolos
```

Output

```

mkdir backup/_backup
cp -a Receitas_Original/_backup/outro_ficheiro.txt backup/_backup/outro_ficheiro.txt
While backuping Receitas_Original/_backup: 0 Errors; 0 Warnings; 0 Updated; 1 Copied (12B); 0 Deleted (0B)
Skipping excluded file or directory: Receitas_Original/Bolos
mkdir backup/Carne
cp -a Receitas_Original/Carne/Cabidela.txt backup/Carne/Cabidela.txt
cp -a Receitas_Original/Carne/Chanfana.txt backup/Carne/Chanfana.txt
While backuping Receitas_Original/Carne: 0 Errors; 0 Warnings; 0 Updated; 2 Copied (2677B); 0 Deleted (0B)
cp -a Receitas_Original/dsdasdasda backup/dsdasdasda
mkdir backup/Massas
mkdir backup/Massas/Italiana
cp -a Receitas_Original/Massas/Italiana/Bolenhesa.txt backup/Massas/Italiana/Bolenhesa.txt
cp -a Receitas_Original/Massas/Italiana/ficheiro_para_eliminar.txt backup/Massas/Italiana/ficheiro_para_eliminar.txt
cp -a Receitas_Original/Massas/Italiana/hbjb backup/Massas/Italiana/hbjb
cp -a Receitas_Original/Massas/Italiana/Lasanha.txt backup/Massas/Italiana/Lasanha.txt
While backuping Receitas_Original/Massas/Italiana: 0 Errors; 0 Warnings; 0 Updated; 4 Copied (1762B); 0 Deleted (0B)
Skipping excluded file or directory: Receitas_Original/Massas/Massas_muito_bom.txt
While backuping Receitas_Original/Massas: 0 Errors; 1 Warnings; 0 Updated; 0 Copied (0B); 0 Deleted (0B)
mkdir backup/src
mkdir backup/src/aaa
cp -a Receitas_Original/src/aaa/prog2.c backup/src/aaa/prog2.c
While backuping Receitas_Original/src/aaa: 0 Errors; 0 Warnings; 0 Updated; 1 Copied (8282B); 0 Deleted (0B)
cp -a Receitas_Original/src/output.txt backup/src/output.txt
cp -a Receitas_Original/src/prog1.c backup/src/prog1.c
cp -a Receitas_Original/src/start.sh backup/src/start.sh
While backuping Receitas_Original/src: 0 Errors; 0 Warnings; 0 Updated; 3 Copied (16596B); 0 Deleted (0B)
cp -a Receitas_Original/ssdasds backup/ssdasds
While backuping Receitas_Original: 0 Errors; 1 Warnings; 0 Updated; 2 Copied (27B); 0 Deleted (0B)

```

Diretoria Original

```

Receitas_Original
├── _backup
│   └── outro_ficheiro.txt
├── Bolos
│   ├── Bolo_Chocolate.txt
│   ├── Bolo_Marmore.txt
│   └── ficheiro_sem_permissoes.txt
├── Carne
│   ├── Cabidela.txt
│   └── Chanfana.txt
├── dsdasdasda
├── Massas
│   └── Italiana
│       ├── Bolenhesa.txt
│       ├── ficheiro_para_eliminar.txt
│       ├── hbjb
│       └── Lasanha.txt
├── Massas_muito_bom.txt
├── src
│   ├── aaa
│   │   ├── prog2.c
│   │   ├── output.txt
│   │   ├── prog1.c
│   │   └── start.sh
│   └── ssdasds

```

Diretoria Backup

```

backup
├── _backup
│   └── outro_ficheiro.txt
├── Carne
│   ├── Cabidela.txt
│   └── Chanfana.txt
├── dsdasdasda
├── Massas
│   └── Italiana
│       ├── Bolenhesa.txt
│       ├── ficheiro_para_eliminar.txt
│       ├── hbjb
│       └── Lasanha.txt
├── src
│   ├── aaa
│   │   ├── prog2.c
│   │   ├── output.txt
│   │   ├── prog1.c
│   │   └── start.sh
│   └── ssdasds

```

Opções - “-r .txt”

Diretoria de trabalho - /home/belito/Documents/GitHub/Backup_SO/Projeto/Receitas_Original

**Diretoria de Backup - /home/belito/Documents/GitHub/Backup_SO/backupup
(Inexistente)**

Output

```
mkdir backupup/ backupup
cp -a Receitas_Original/ backupup/outro_ficheiro.txt backupup/ backupup/outro_ficheiro.txt
While backuping Receitas_Original/ backupup: 0 Errors; 0 Warnings; 0 Updated; 1 Copied (12B); 0 Deleted (0B)
mkdir backupup/Bolos
cp -a Receitas_Original/Bolos/Bolo_Chocolate.txt backupup/Bolos/Bolo_Chocolate.txt
cp -a Receitas_Original/Bolos/Bolo_Marmore.txt backupup/Bolos/Bolo_Marmore.txt
cp -a Receitas_Original/Bolos/ficheiro_sem_permissoes.txt backupup/Bolos/ficheiro_sem_permissoes.txt
While backuping Receitas_Original/Bolos: 0 Errors; 0 Warnings; 0 Updated; 3 Copied (4361B); 0 Deleted (0B)
mkdir backupup/Carne
cp -a Receitas_Original/Carne/Cabidela.txt backupup/Carne/Cabidela.txt
cp -a Receitas_Original/Carne/Chanfana.txt backupup/Carne/Chanfana.txt
While backuping Receitas_Original/Carne: 0 Errors; 0 Warnings; 0 Updated; 2 Copied (2677B); 0 Deleted (0B)
WARNING: Skipping Receitas_Original/dsdsdasda: doesnt match the provided expression.
mkdir backupup/Massas
mkdir backupup/Massas/Italiana
cp -a Receitas_Original/Massas/Italiana/Bolenhesa.txt backupup/Massas/Italiana/Bolenhesa.txt
cp -a Receitas_Original/Massas/Italiana/ficheiro_para_eliminar.txt backupup/Massas/Italiana/ficheiro_para_eliminar.txt
WARNING: Skipping Receitas_Original/Massas/Italiana/hbjb: doesnt match the provided expression.
cp -a Receitas_Original/Massas/Italiana/Lasanha.txt backupup/Massas/Italiana/Lasanha.txt
While backuping Receitas_Original/Massas/Italiana: 0 Errors; 1 Warnings; 0 Updated; 3 Copied (1761B); 0 Deleted (0B)
cp -a Receitas_Original/Massas/Massas_muito_bom.txt backupup/Massas/Massas_muito_bom.txt
While backuping Receitas_Original/Massas: 0 Errors; 0 Warnings; 0 Updated; 1 Copied (252B); 0 Deleted (0B)
mkdir backupup/src
mkdir backupup/src/aaa
WARNING: Skipping Receitas_Original/src/aaa/prog2.c: doesnt match the provided expression.
While backuping Receitas_Original/src/aaa: 0 Errors; 1 Warnings; 0 Updated; 0 Copied (0B); 0 Deleted (0B)
cp -a Receitas_Original/src/output.txt backupup/src/output.txt
WARNING: Skipping Receitas_Original/src/prog1.c: doesnt match the provided expression.
WARNING: Skipping Receitas_Original/src/start.sh: doesnt match the provided expression.
While backuping Receitas_Original/src: 0 Errors; 2 Warnings; 0 Updated; 1 Copied (10B); 0 Deleted (0B)
WARNING: Skipping Receitas_Original/ssdsds: doesnt match the provided expression.
While backuping Receitas_Original: 0 Errors; 2 Warnings; 0 Updated; 0 Copied (0B); 0 Deleted (0B)
```

Diretoria Original

```
Receitas_Original
├── backup
│   └── outro_ficheiro.txt
├── Bolos
│   ├── Bolo_Chocolate.txt
│   ├── Bolo_Marmore.txt
│   └── ficheiro_sem_permissoes.txt
├── Carne
│   ├── Cabidela.txt
│   └── Chanfana.txt
├── dsdsdasda
├── Massas
│   └── Italiana
│       ├── Bolenhesa.txt
│       ├── ficheiro_para_eliminar.txt
│       ├── hbjb
│       └── Lasanha.txt
│       └── Massas_muito_bom.txt
├── src
│   ├── aaa
│   │   ├── prog2.c
│   │   ├── output.txt
│   │   ├── prog1.c
│   │   └── start.sh
│   └── ssdsds
```

Diretoria Backup

```
backupup
├── backup
│   └── outro_ficheiro.txt
├── Bolos
│   ├── Bolo_Chocolate.txt
│   ├── Bolo_Marmore.txt
│   └── ficheiro_sem_permissoes.txt
├── Carne
│   ├── Cabidela.txt
│   └── Chanfana.txt
├── Massas
│   └── Italiana
│       ├── Bolenhesa.txt
│       ├── ficheiro_para_eliminar.txt
│       ├── Lasanha.txt
│       └── Massas_muito_bom.txt
├── src
│   ├── aaa
│   └── output.txt
```

Opções - “-c -b ex.txt -r .txt”

Diretoria de trabalho - /home/belito/Documents/GitHub/Backup_SO/Projeto/Receitas_Original

Diretoria de Backup - /home/belito/Documents/GitHub/Backup_SO/backupup
(Inexistente)

Output

```
mkdir backupup/_backup
cp -a Receitas_Original/_backup/outro_ficheiro.txt backupup/_backup/outro_ficheiro.txt
While backuping Receitas_Original/_backup: 0 Errors; 0 Warnings; 0 Updated; 1 Copied (12B); 0 Deleted (0B)
Skipping excluded file or directory: Receitas_Original/Bolos
mkdir backupup/Carne
cp -a Receitas_Original/Carne/Cabidela.txt backupup/Carne/Cabidela.txt
cp -a Receitas_Original/Carne/Chanfana.txt backupup/Carne/Chanfana.txt
While backuping Receitas_Original/Carne: 0 Errors; 0 Warnings; 0 Updated; 2 Copied (2677B); 0 Deleted (0B)
WARNING: Skipping Receitas_Original/dsdasdasda: doesnt match the provided expression.
mkdir backupup/Massas
mkdir backupup/Massas/Italiana
cp -a Receitas_Original/Massas/Italiana/Bolenhesa.txt backupup/Massas/Italiana/Bolenhesa.txt
cp -a Receitas_Original/Massas/Italiana/ficheiro_para_eliminar.txt backupup/Massas/Italiana/ficheiro_para_eliminar.txt
WARNING: Skipping Receitas_Original/Massas/Italiana/hbjb: doesnt match the provided expression.
cp -a Receitas_Original/Massas/Italiana/Lasanha.txt backupup/Massas/Italiana/Lasanha.txt
While backuping Receitas_Original/Massas/Italiana: 0 Errors; 1 Warnings; 0 Updated; 3 Copied (1761B); 0 Deleted (0B)
Skipping excluded file or directory: Receitas_Original/Massas/Massas_muito_bom.txt
While backuping Receitas_Original/Massas: 0 Errors; 1 Warnings; 0 Updated; 0 Copied (0B); 0 Deleted (0B)
mkdir backupup/src
mkdir backupup/src/aaa
WARNING: Skipping Receitas_Original/src/aaa/prog2.c: doesnt match the provided expression.
While backuping Receitas_Original/src/aaa: 0 Errors; 1 Warnings; 0 Updated; 0 Copied (0B); 0 Deleted (0B)
cp -a Receitas_Original/src/output.txt backupup/src/output.txt
WARNING: Skipping Receitas_Original/src/progl.c: doesnt match the provided expression.
WARNING: Skipping Receitas_Original/src/start.sh: doesnt match the provided expression.
While backuping Receitas_Original/src: 0 Errors; 2 Warnings; 0 Updated; 1 Copied (10B); 0 Deleted (0B)
WARNING: Skipping Receitas_Original/ssdasds: doesnt match the provided expression.
While backuping Receitas_Original: 0 Errors; 3 Warnings; 0 Updated; 0 Copied (0B); 0 Deleted (0B)
```

Descrição – Teríamos permissões a alguns ficheiros da diretoria original de maneira a verificar se o programa continuava a execução, se o erro era apresentado e se o erros eram incrementados.

Diretoria de trabalho - /home/belito/Documents/GitHub/Backup_SO/Projeto/Receitas_Original

Diretoria de Backup - /home/belito/Documents/GitHub/Backup_SO/backupup
(Inexistente)

```
belito@belitopc:~/Documents/GitHub/Backup_SO/Projeto/Receitas_Original$ chmod -rwx dsdasdasda
belito@belitopc:~/Documents/GitHub/Backup_SO/Projeto/Receitas_Original$ cd Bolos
belito@belitopc:~/Documents/GitHub/Backup_SO/Projeto/Receitas_Original/Bolos$ chmod -rwx Bolo Chocolate.txt
```

Output


```

mkdir backup/ backup
cp -a Receitas_Original/_backup/outro_ficheiro.txt backup/_backup/outro_ficheiro.txt
While backuping Receitas_Original/_backup: 0 Errors; 0 Warnings; 0 Updated; 1 Copied (19B); 0 Deleted (0B)
mkdir backup/Bolos
cp -a Receitas_Original/Bolos/Bolo_Chocolate.txt backup/Bolos/Bolo_Chocolate.txt
cp: cannot open '/home/belito/Documents/GitHub/Backup_SO/Projeto/Receitas_Original/Bolos/Bolo_Chocolate.txt' for reading: Permission denied
cp -a Receitas_Original/Bolos/Bolo_Marmore.txt backup/Bolos/Bolo_Marmore.txt
cp -a Receitas_Original/Bolos/ficheiro_sem_permissoes.txt backup/Bolos/ficheiro_sem_permissoes.txt
While backuping Receitas_Original/Bolos: 1 Errors; 0 Warnings; 0 Updated; 2 Copied (1773B); 0 Deleted (0B)
mkdir backup/Carne
cp -a Receitas_Original/Carne/Cabidela.txt backup/Carne/Cabidela.txt
cp -a Receitas_Original/Carne/Chanfana.txt backup/Carne/Chanfana.txt
While backuping Receitas_Original/Carne: 0 Errors; 0 Warnings; 0 Updated; 2 Copied (2689B); 0 Deleted (0B)
cp -a Receitas_Original/dsdasdasda backup/dsdasdasda
cp: cannot open '/home/belito/Documents/GitHub/Backup_SO/Projeto/Receitas_Original/dsdasdasda' for reading: Permission denied
mkdir backup/Massas
mkdir backup/Massas/Italiana
cp -a Receitas_Original/Massas/Italiana/Bolenhesa.txt backup/Massas/Italiana/Bolenhesa.txt
cp -a Receitas_Original/Massas/Italiana/ficheiro_para_eliminar.txt backup/Massas/Italiana/ficheiro_para_eliminar.txt
cp -a Receitas_Original/Massas/Italiana/hbjb backup/Massas/Italiana/hbjb
cp -a Receitas_Original/Massas/Italiana/Lasanha.txt backup/Massas/Italiana/Lasanha.txt
While backuping Receitas_Original/Massas/Italiana: 0 Errors; 0 Warnings; 0 Updated; 4 Copied (1762B); 0 Deleted (0B)
cp -a Receitas_Original/Massas/Massas_muito_bom.txt backup/Massas/Massas_muito_bom.txt
While backuping Receitas_Original/Massas: 0 Errors; 0 Warnings; 0 Updated; 1 Copied (252B); 0 Deleted (0B)
mkdir backup/src
mkdir backup/src/aaa
cp -a Receitas_Original/src/aaa/prog2.c backup/src/aaa/prog2.c
While backuping Receitas_Original/src/aaa: 0 Errors; 0 Warnings; 0 Updated; 1 Copied (8282B); 0 Deleted (0B)
cp -a Receitas_Original/src/output.txt backup/src/output.txt
cp -a Receitas_Original/src/prog1.c backup/src/prog1.c
cp -a Receitas_Original/src/start.sh backup/src/start.sh
While backuping Receitas_Original/src: 0 Errors; 0 Warnings; 0 Updated; 3 Copied (16596B); 0 Deleted (0B)
cp -a Receitas_Original/ssdasds backup/ssdasds
While backuping Receitas_Original: 1 Errors; 0 Warnings; 0 Updated; 1 Copied (12B); 0 Deleted (0B)

```

Diretoria Original

```

Receitas_Original
├── backup
│   └── outro_ficheiro.txt
├── Bolos
│   ├── Bolo_Chocolate.txt
│   ├── Bolo_Marmore.txt
│   └── ficheiro_sem_permissoes.txt
├── Carne
│   ├── Cabidela.txt
│   └── Chanfana.txt
├── dsdasdasda
├── Massas
│   ├── Italiana
│   │   ├── Bolenhesa.txt
│   │   ├── ficheiro_para_eliminar.txt
│   │   ├── hbjb
│   │   └── Lasanha.txt
│   └── Massas_muito_bom.txt
├── src
│   ├── aaa
│   │   ├── prog2.c
│   │   ├── output.txt
│   │   ├── prog1.c
│   │   └── start.sh
│   └── ssdasds

```

Diretoria Backup

```

backup
├── backup
│   └── outro_ficheiro.txt
├── Bolos
│   ├── Bolo_Marmore.txt
│   └── ficheiro_sem_permissoes.txt
├── Carne
│   ├── Cabidela.txt
│   └── Chanfana.txt
├── Massas
│   ├── Italiana
│   │   ├── Bolenhesa.txt
│   │   ├── ficheiro_para_eliminar.txt
│   │   ├── hbjb
│   │   └── Lasanha.txt
│   └── Massas_muito_bom.txt
├── src
│   ├── aaa
│   │   ├── prog2.c
│   │   ├── output.txt
│   │   ├── prog1.c
│   │   └── start.sh
│   └── ssdasds

```

Descrição – Eliminar ficheiros da diretoria original de maneira a verificar se o programa continua a execução, se o erro era apresentado e se o ficheiros apagados eram incrementados

Diretoria de trabalho - /home/belito/Documents/GitHub/Backup_SO/Projeto/Receitas_Original

Diretoria de Backup - /home/belito/Documents/GitHub/Backup_SO/backup

```

belito@belitopc:~/Documents/GitHub/Backup_SO/Projeto/Receitas_Original$ rm dsdadada
belito@belitopc:~/Documents/GitHub/Backup_SO/Projeto/Receitas_Original$ cd Bolos
belito@belitopc:~/Documents/GitHub/Backup_SO/Projeto/Receitas_Original/Bolos$ rm Bolo_Chocolate.txt

```

Output

```
While backuping Receitas_Original/_backup: 0 Errors; 0 Warnings; 0 Updated; 0 Copied (0B); 0 Deleted (0B)
rm backup/Bolos/Bolo_Chocolate.txt
While backuping Receitas_Original/Bolos: 0 Errors; 0 Warnings; 0 Updated; 0 Copied (0B); 1 Deleted (2588B)
While backuping Receitas_Original/Carne: 0 Errors; 0 Warnings; 0 Updated; 0 Copied (0B); 0 Deleted (0B)
While backuping Receitas_Original/Massas/Italiana: 0 Errors; 0 Warnings; 0 Updated; 0 Copied (0B); 0 Deleted (0B)
While backuping Receitas_Original/Massas: 0 Errors; 0 Warnings; 0 Updated; 0 Copied (0B); 0 Deleted (0B)
While backuping Receitas_Original/src/aaa: 0 Errors; 0 Warnings; 0 Updated; 0 Copied (0B); 0 Deleted (0B)
While backuping Receitas_Original/src: 0 Errors; 0 Warnings; 0 Updated; 0 Copied (0B); 0 Deleted (0B)
rm backup/dsdadada
While backuping Receitas_Original: 0 Errors; 0 Warnings; 0 Updated; 0 Copied (0B); 1 Deleted (1B)
```

Diretoria Original

```
Receitas_Original
├── _backup
│   └── outro_ficheiro.txt
├── Bolos
│   ├── Bolo_Marmore.txt
│   └── ficheiro_sem_permissoes.txt
├── Carne
│   ├── Cabidela.txt
│   └── Chanfana.txt
├── Massas
│   ├── Italiana
│   │   ├── Bolenhesa.txt
│   │   ├── ficheiro_para_eliminar.txt
│   │   ├── hbjb
│   │   └── Lasanha.txt
│   └── Massas_muito_bom.txt
├── src
│   ├── aaa
│   │   └── prog2.c
│   ├── output.txt
│   ├── prog1.c
│   └── start.sh
└── ssdasds
```

Diretoria Backup

```
backup
├── _backup
│   └── outro_ficheiro.txt
├── Bolos
│   ├── Bolo_Marmore.txt
│   └── ficheiro_sem_permissoes.txt
├── Carne
│   ├── Cabidela.txt
│   └── Chanfana.txt
├── Massas
│   ├── Italiana
│   │   ├── Bolenhesa.txt
│   │   ├── ficheiro_para_eliminar.txt
│   │   ├── hbjb
│   │   └── Lasanha.txt
│   └── Massas_muito_bom.txt
├── src
│   ├── aaa
│   │   └── prog2.c
│   ├── output.txt
│   ├── prog1.c
│   └── start.sh
└── ssdasds
```

Descrição – Modificar ficheiros da diretoria original de maneira a verificar se o programa continua a execução e se o ficheiros atualizados eram incrementados.

Diretoria de trabalho - /home/belito/Documents/GitHub/Backup_SO/Projeto/Receitas_Original

Diretoria de Backup - /home/belito/Documents/GitHub/Backup_SO/backup

Modificamos: Chanfana.txt e ssdasds

Output

```
While backuping Receitas_Original/ backup: 0 Errors; 0 Warnings; 0 Updated; 0 Copied (0B); 0 Deleted (0B)
While backuping Receitas_Original/Bolos: 0 Errors; 0 Warnings; 0 Updated; 0 Copied (0B); 0 Deleted (0B)
cp -a Receitas_Original/Carne/Chanfana.txt backupup/Carne/Chanfana.txt
While backuping Receitas_Original/Carne: 0 Errors; 0 Warnings; 1 Updated; 0 Copied (0B); 0 Deleted (0B)
While backuping Receitas_Original/Massas/Italiana: 0 Errors; 0 Warnings; 0 Updated; 0 Copied (0B); 0 Deleted (0B)
While backuping Receitas_Original/Massas: 0 Errors; 0 Warnings; 0 Updated; 0 Copied (0B); 0 Deleted (0B)
While backuping Receitas_Original/src/aaa: 0 Errors; 0 Warnings; 0 Updated; 0 Copied (0B); 0 Deleted (0B)
While backuping Receitas_Original/src: 0 Errors; 0 Warnings; 0 Updated; 0 Copied (0B); 0 Deleted (0B)
cp -a Receitas_Original/ssdasds backupup/ssdasds
While backuping Receitas_Original: 0 Errors; 0 Warnings; 1 Updated; 0 Copied (0B); 0 Deleted (0B)
```

-Backup_check.sh

Descrição – Modificar ficheiros da diretoria original de maneira a verificar se o programa dava o output correspondente.

Diretoria de trabalho - /home/belito/Documents/GitHub/Backup_SO/Projeto/Receitas_Original

Diretoria de Backup - /home/belito/Documents/GitHub/Backup_SO/backupup

Modificamos: Chanfana.txt e ssdasds

Output

```
Receitas_Original/Carne/Chanfana.txt backupup/Carne/Chanfana.txt differ.
Receitas_Original/ssdasds backupup/ssdasds differ.
```

Conclusão

Pensamos ter obtido um resultado satisfatório no desenvolvimento do projeto, pois cumpre com as normas dadas e com o que nós estávamos á espera. Após este projeto ficamos claramente com o nosso conhecimento de bash um pouco mais consolidado.

Webgrafia

- <https://medium.com/@shigmas/path-manipulation-in-the-shell-6f02dee65098>
- <https://www.linuxjournal.com/content/how-count-files-directory-linux>
- <https://www.geeksforgeeks.org/getopts-command-in-linux-with-examples/>
- <https://phoenixnap.com/kb/bash-eval>
- <https://medium.com/@linuxadminhacks/understanding-ifs-in-bash-scripting-3c67a39661e9>
- <https://unix.stackexchange.com/questions/16640/how-can-i-get-the-size-of-a-file-in-a-bash-script>
- <https://stackoverflow.com/questions/16483119/an-example-of-how-to-use-getopts-in-bash>