Ciências ULisboa | Informática

Sistemas Operativos

2022/23

Trabalho de projeto – 1ª Fase

Objetivo

Realização de um trabalho de programação em Python envolvendo a criação de processos/*threads* e a comunicação entre processos/*threads.*

Introdução

O comando *grep* pesquisa as linhas dos ficheiros que contêm determinada palavra ou expressão regular. Uma dada palavra a procurar pode aparecer isolada dentro de uma linha (i.e., está separada das restantes palavras dentro da linha de texto) ou fazer parte de outra palavra. Por exemplo, nas duas frases seguintes: "*O Sistema Operativo utiliza o hardware*" e "*O Sistema Operativo satisfaz os pedidos do utilizador*"; se procurarmos a palavra "*utiliza*", o comando *grep* devolve ambas as linhas. Este comando, quando usado com várias palavras a pesquisar e com um conjunto alargado de ficheiros, apresenta alguns problemas de desempenho.

Com este trabalho pretende-se desenvolver o comando `pgrepwc` (parallel grep with counting), uma versão do *grep* com funcionalidades acrescidas e que paraleliza as procuras. O comando escreverá na saída as linhas de texto que contêm a palavra a pesquisar, a contagem das linhas resultantes e o número de ocorrências da palavra a pesquisar (p. ex., uma linha pode conter mais do que uma ocorrência da palavra a pesquisar).

Descrição do trabalho

Pretende-se que os alunos implementem o comando `pgrepwc` descrito de seguida:

NOME

`pgrepwc` – pesquisa uma determinada palavra num ou mais ficheiros, escrevendo na saída as linhas de texto que contêm pelo menos uma ocorrência isolada da palavra. O comando também pode escrever na saída o número total de ocorrências isoladas da palavra, bem como o número total de linhas onde esta foi encontrada. O nível e a forma de paralelização da procura e contagem de ocorrências são determinados pelo utilizador.

SINOPSE

```
./pgrepwc [-c] [-l] [-p n] [-t] [-e] palavra {ficheiros}
```

DESCRIÇÃO

`-c`: opção que permite contar o número total de ocorrências isoladas da palavra a pesquisar.

`-l`: opção que permite obter o número total de linhas que contêm uma ou mais ocorrências da palavra a pesquisar.

`-p n`: opção que permite definir o nível de paralelização `n` do comando (ou seja, o número de processos (filhos)/*threads* que são utilizados para efetuar a pesquisa e as contagens). Por omissão, deve ser utilizado apenas um processo (o processo pai) para realizar a pesquisa e contagens.

`-t`: opção que indica que se deve utilizar o programa com *threads*. Caso contrário, será usado o programa com processos.

`-e`: opção que permite ativar o modo de paralelização especial. Se forem especificados vários ficheiros, esta opção é ignorada.

`palavra`: a palavra a pesquisar no conteúdo do(s) ficheiro(s).

`ficheiros`: podem ser dados um ou mais ficheiros, sobre os quais é efetuada a pesquisa e contagem. Caso não sejam dados ficheiros na linha de comandos, estes devem ser lidos de *stdin* (o comando no início da sua execução pedirá ao utilizador quais são os ficheiros a processar).

Inicialmente, após a validação das opções do comando, o processo pai deve criar os processos filhos/*threads* definidos pelo nível de paralelização do comando (valor `n`). Estes processos/*threads* pesquisam as palavras nos ficheiros, contam as ocorrências das palavras e o número de linhas em que estas foram encontradas nos ficheiros, e escrevem os resultados (linhas encontradas e contagens) na saída (*stdout*). Os resultados das pesquisas e contagens são escritos para *stdout* de forma não intercalada, ou seja, os resultados de cada processo/*thread* são apresentados sequencialmente, sem serem intercalados com os resultados dos outros processos/*threads*.

A paralelização do trabalho de pesquisa e contagem é coordenada pelo processo pai. Esta paralelização pode ser feita de duas formas:

a) No caso normal, cada ficheiro é atribuído pelo processo pai a um único processo/*thread*, não havendo assim divisão do conteúdo de um ficheiro por vários processos/*threads*. Neste caso, se o valor de `n` for superior ao número de ficheiros, o comando (o processo pai) redefine-o automaticamente para o número de ficheiros, ou seja, cria tantos processos/*threads* quantos os ficheiros a pesquisar. Se existirem mais ficheiros do que processos/*threads*, o processo pai deverá decidir inicialmente como distribuir os ficheiros, tentando ser o mais equitativo possível (em termos do número de ficheiros atribuídos a cada processo/*thread*);

b) No caso especial de ser indicado apenas um ficheiro a pesquisar e de ser indicada a opção `-e`, então a paralelização será feita pelo processo pai de forma que o conteúdo do ficheiro indicado seja dividido pelos `n` processos/*threads* que estiverem disponíveis. O processo pai terá de saber quantas linhas tem o ficheiro para poder fazer uma divisão tão equitativa quanto possível.

No final, e independentemente da forma de paralelização, o processo pai terá de escrever para *stdout* o número total de ocorrências das palavras e/ou de linhas encontradas, de acordo com a opção especificada (`-c` e/ou `-l`).

Os alunos têm de implementar duas soluções: uma com processos e outra com *threads*.

Desafios

- Como garantir que a pesquisa e a contagem são efetuadas em todos os ficheiros uma e apenas uma vez?
- Como garantir que os resultados para *stdout* são escritos de forma não intercalada?
- Como passar a informação necessária ao processo pai de modo a ser possível calcular o número total de ocorrência das palavras a pesquisar ou linhas encontradas?
- Como concretizar a divisão de um ficheiro em várias partes, e como atribuir cada parte a um processo/*thread*?

Ficheiros iniciais e de teste

Juntamente com o enunciado do projeto, serão disponibilizados um ficheiro ZIP (`grupoXX.zip`) com a estrutura inicial do projeto e quatro ficheiros de texto que servirão para os alunos testarem as duas soluções do comando `pgrepwc`. Os alunos terão de descarregar estes ficheiros para a sua máquina. Não os deverão abrir no Moodle, principalmente o ficheiro `file1.txt` por este ser grande (~500 MB).

Entrega

A entrega do trabalho é realizada da seguinte forma:

- Os grupos inscrevem-se atempadamente, de acordo com as regras afixadas para o efeito, no Moodle.
- Colocar os ficheiros `pgrepwc`, `pgrepwc_processos.py` e `pgrepwc_threads.py` do projeto numa diretoria cujo nome deve seguir exatamente o padrão **grupoXX** (por exemplo grupo01 ou grupo23). Juntamente com os três ficheiros, incluir um ficheiro de texto README.txt (não é .pdf nem .rtf nem .doc nem .docx) que deve conter:

1. A identificação dos elementos do grupo;
2. Exemplos de chamadas do comando `pgrepwc`;
3. As limitações da implementação;
4. A abordagem usada para a divisão dos ficheiros pelos processos/*threads*;
5. Outras informações que acharem pertinente sobre a implementação do projeto.

- A diretoria será incluída num ficheiro ZIP cujo nome deve seguir exatamente o padrão **grupoXX.zip**. Esse ficheiro deverá ser submetido no Moodle (um por grupo).

De notar que **a entrega deve conter apenas a diretoria com o ficheiro `pgrepwc`, os dois ficheiros .py e o ficheiro README.txt, pois qualquer outro ficheiro será ignorado**.

Se não se verificar algum destes requisitos o trabalho é considerado não entregue.

Não serão aceites trabalhos entregues por mail nem por qualquer outro meio não definido nesta secção.

Prazo de entrega

O trabalho deve ser entregue até dia **06 de novembro de 2022 (domingo) às 23:59h**.

Avaliação dos trabalhos

A avaliação do trabalho será realizada:

(1) pelos alunos, pelo preenchimento do formulário de contribuição de cada aluno no desenvolvimento do projeto. O formulário será disponibilizado no Moodle e preenchido após a entrega do projeto.

(2) pelo corpo docente, sobre dois conjuntos de ficheiros de texto: os ficheiros de teste disponibilizados aos alunos e outros somente usados pelos docentes.

Para além dos testes a efetuar, os seguintes parâmetros serão avaliados: funcionalidade, estrutura, desempenho, algoritmia, comentários, clareza do código, validação dos parâmetros de entrada e tratamento de erros.

Divulgação dos resultados

A data prevista da divulgação dos resultados da avaliação dos trabalhos é 25 de novembro de 2022.