

**ATENÇÃO:** Responda às questões **1** e **2** num conjunto de folhas e às questões **3**, **4** e **5** noutro conjunto.

1. [6] Considere o seguinte programa fonte, compilado para o executável **ex1**:

```
int main(int argc, char * argv[]) {  
    int base = atoi(argv[1]); // converte string com dígitos para inteiro  
    for (int i = 0; i < 4; ++i) printf("%d\n", base + i);  
    return 0;  
}
```

- a. [2] Indique, justificando, o que é afixado no *standard output* e qual o conteúdo dos ficheiros `output*.txt`, com a execução de cada uma das seguintes linhas de comandos pelo *shell*:
- `ex1 18 | grep 1 > output1.txt` ← também para a alínea b
- `ex1 77 | tee output2.txt | grep 8`
- b. [4] Escreva um programa em C que execute especificamente a primeira linha de comandos indicada na alínea anterior, usando diretamente a API POSIX para criar/abrir ficheiro(s), lançar os processos necessários, ligá-los da mesma forma que o *shell* e aguardar pela sua conclusão.
2. [4] Durante a tradução de **um** endereço virtual por um processador *x86-64* foram consultadas quatro tabelas de páginas, pela ordem indicada, nos seguintes índices:
- 1º nível (PML4) índice 17 (PTE com bit R/W = 1)
  - 2º nível (PDP) índice 1 (PTE com bit R/W = 1)
  - 3º nível (PD) índice 9 (PTE com bit R/W = 0)
  - 4º nível (PT) índice 3 (PTE com bit R/W = 1)
- a. [2] Qual é o endereço virtual mais baixo que poderia estar a ser traduzido? Apresente os cálculos.
- b. [2] A tabela de 4º nível (PT) só é referida pelo índice 9 da tabela de 3º nível (PD) indicada. Todas as suas entradas estão preenchidas (bit P = 1) e mapeadas para endereços físicos diferentes. Um quarto ( $\frac{1}{4}$ ) das entradas têm o bit R/W = 0 e três quartos ( $\frac{3}{4}$ ) têm R/W = 1. Em quantos endereços virtuais cobertos por esta *page table* (PT) é possível ler, mas proibido escrever? Justifique.
3. [4.5] Considere o seguinte código fonte de uma biblioteca (ficheiro com extensão `.so` ou *shared object*):

```
const int BASE[] = {0,11,22,33};  
int accum = 1;  
int func(int idx) { accum += idx; return BASE[idx%4] + accum; }
```

- a. [2.5] Tendo em conta apenas o código apresentado, que alterações se pode prever que terá o espaço de endereçamento de um processo que carregue esta biblioteca com `dlopen`?
- b. [2] Se, simultaneamente, quatro processos distintos carregarem esta biblioteca e executarem pelo menos uma vez a função `func`, indique, justificando, qual espera que seja o *proportional set size* (Pss) das regiões relevantes relativas à biblioteca logo após os retornos de `func`?

(continua)

4. [2.5] No âmbito dos processadores *x86-64*, o que são **exceções** e que relação têm com níveis de privilégio? Na sua resposta inclua pelo menos dois exemplos de ocorrências de exceções, um para um caso efetivo de erro e outro para uma situação que, apesar de ser tratada como exceção pelo processador, pode não corresponder a um erro, do ponto de vista do sistema operativo.
5. [3] Em cada um dos casos seguintes, qual é o *backing storage* para as páginas físicas de memória referidas?
- a. [0.6] Página pertencente ao mapeamento da secção de código (*.text*) de um ficheiro executável.
  - b. [0.6] Página modificada, pertencente ao mapeamento da secção de dados inicializados de um executável.
  - c. [0.6] Página modificada, pertencente ao mapeamento da secção de dados não-inicializados de uma biblioteca (ficheiro com extensão *.so* ou *shared object*).
  - d. [0.6] Página não modificada, pertencente ao mapeamento PRIVATE de um ficheiro comum.
  - e. [0.6] Página não modificada, pertencente ao mapeamento de uma região anónima partilhada (MAP\_SHARED | MAP\_ANONYMOUS).

Duração: 1 hora e 15 minutos

ISEL, 8 de novembro de 2022