



1º Trabalho Prático – Processos/Tarefas

Objectivo:

Desenvolvimento de aplicações no sistema Operativo Windows, usando a API Win32; análise e desenvolvimento de aplicações recorrendo a multiprocessamento tirando conclusões, relativamente, a vantagens/desvantagens em diversos cenários de execução. Escrita de código independente do tipo de codificação de caracteres utilizado (ASCII/Unicode).

Análise dos exemplos

Analise os exemplos disponíveis na página da disciplina, no moodle (ver figura 1), sobre a criação de processos e de tarefas na WIN32 API.



Figura 1 – Exemplos referentes à criação de processos e tarefas

Utilize a ferramenta *ProcessExplorer*¹, apresentada na figura 2 (disponível na página da disciplina), para analisar os recursos utilizados por cada aplicação, como por exemplo, número de threads, prioridade dos processo e tarefas, handles, módulos constituintes de um processo, etc.

Como complemento ao estudo e preparação para a realização do trabalho prático sugerem-se as seguintes alterações aos exemplos:

- Altere o exemplo 04-CriacaoTarefas de modo a permitir receber nos seus argumentos o número de tarefas a lançar e o tempo de execução que estas deverão demorar (assuma que o tempo de execução é efectuado através da chamada de sistema `Sleep(DWORD)`).

¹ www.sysinternals.com

- Experimente executar o exemplo 04-CriacaoTarefas através do exemplo 03-CriacaoProcessos.

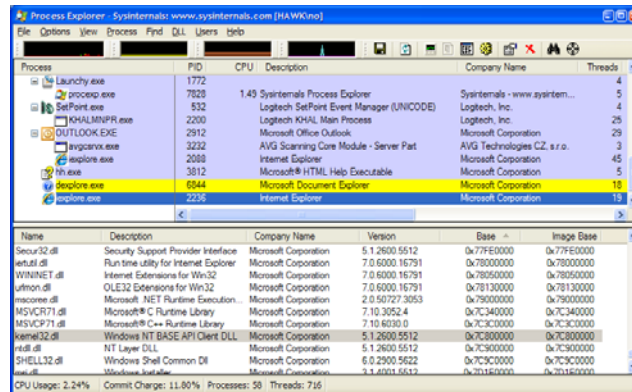


Figura 2 - Ferramenta *Process Explorer* da Sysinternals

Cálculo do número PI com múltiplas tarefas

Pretende-se uma aplicação Win32, em modo consola, com múltiplas tarefas para determinar o valor do número PI. O método a utilizar baseia-se na geração de pontos aleatórios (número de experiências) contidos num quadrado de dimensão unitária contabilizando o número de pontos que pertencem ao maior círculo inscrito no quadrado. Sabendo que a área do círculo (πr^2) é igual a $\pi/4$ para um quadrado com $L=1$ em que a área ($L \times L$) é igual a 1 tem-se que a razão entre o número de pontos que pertencem ao círculo e o número total de experiências é um valor aproximado a $\pi/4$.

Uma estratégia a seguir consiste na existência de várias tarefas dividindo o número de experiências entre si. Assim, cada tarefa gera pontos aleatórios e contabiliza o número de pontos contidos no círculo. Depois de todas as tarefas terminarem somam-se os contadores parciais de cada tarefa e divide-se pelo número de experiências obtendo assim um valor aproximado a $\pi/4$.

Considere a possibilidade de indicar à aplicação, através da linha de comando, o número de tarefas e o número de experiências a utilizar. Na figura 3 seguinte mostra-se um exemplo de cálculo com base em duas tarefas e um total de 400 mil experiências.

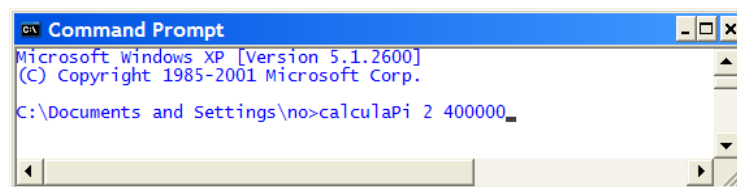


Figura 3 – Exemplo da invocação do programa de cálculo do valor de PI com 400 mil experiências

Avalie o tempo total de execução variando o número de experiências e o número de tarefas. Experimente obter resultados, executando a aplicação, em diferentes arquitecturas *hardware*. Na apresentação dos resultados, não se esqueça de identificar as várias arquitecturas utilizadas nos ensaios, como por exemplo, o tipo de processador.

Se alterar as prioridades das tarefas do cálculo do PI que implicações existem tanto ao nível do cálculo como ao nível de todo o sistema

NOTAS:

- Para gerar números aleatórios utiliza a função `rand` (ver MSDN para mais detalhes)
- Para obter os tempos de execução poderá recorrer às primitivas: `QueryPerformanceCounter`, `GetTickCount`, `timeGetTime`.

Criação de processos e sincronização com a sua terminação

Com base na aplicação desenvolvida, na alínea anterior, realize uma versão multiprocesso em que cada fio de execução paralelo se execute num processo distinto. O processo principal (pai) deverá criar os processos auxiliares, parametrizando-os, convenientemente, com o número de experiências. O processo principal deverá aguardar que os processos filhos terminem e apresentar o resultado final.

Exercícios Teóricos

1. Explique a necessidade da existência de instruções privilegiadas no suporte à realização do sistema operativo.
2. Nos sistemas operativos multiprogramados e multiutilizadores uma das acções realizadas, pelo núcleo, é a comutação de contexto (context switch). Diga em que consiste e em que circunstância é realizada.
3. No decorrer da execução de uma tarefa esta vai mudando de estado. Esses estados são definidos pela actividade corrente das tarefas e atribuídos pelo sistema operativo. Descreva, genericamente, os principais estados em que uma tarefa se pode encontrar assim como as possíveis transições entre estados.
4. Nos sistemas operativos actuais existem, geralmente, os conceitos de tarefa e processo. Assim, o programador tem a possibilidade de modular as suas aplicações com base em múltiplos processos ou em múltiplas tarefas. Aponte as principais vantagens/desvantagens pela adopção de cada uma das alternativas.
5. O sistema operativo multitarefa possui múltiplas tarefas em competição pelo(s) CPU('s). A decisão de qual a próxima tarefa a executar-se é responsabilidade do escalonador (*scheduler*). Num sistema que utilize um escalonamento com base no nível de prioridade das tarefas indique o que entende por *preemptive scheduling* e em que consiste o conceito de *starvation*.

Entrega do trabalho de grupo

A entrega deverá ser feita até ao dia **25 de Outubro de 2010**.

A entrega do trabalho **realiza-se, exclusivamente, na página da turma no Moodle**. A entrega do trabalho é constituída pelo relatório, onde lista e explica a sua solução e resultados observados, e as soluções do *Visual Studio* para que possam ser testadas na discussão. Nas directorias das soluções do *Visual Studio* existe um ficheiro com a extensão *ncb* (que contém informação de suporte ao *Intellisense*) que deve ser eliminado de forma a diminuir a dimensão da solução a ser submetida. Deve, igualmente, eliminar o conteúdo das directorias *Debug* e/ou *Release* da sua solução.

Todos os elementos que compõem o trabalho (relatório, código, etc.) deverão ser entregues num ficheiro comprimido do tipo *Zip* ou *Rar*. O relatório deverá ser entregue no formato *pdf*².

Bom trabalho,

Nuno Oliveira

² Poderá criar ficheiros em pdf com o seguinte utilitário grátis: <http://www.primopdf.com/>