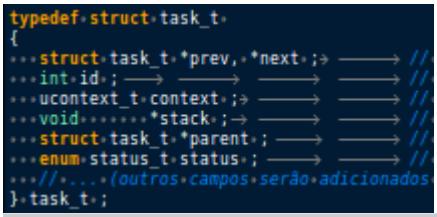


Carlos Maziero

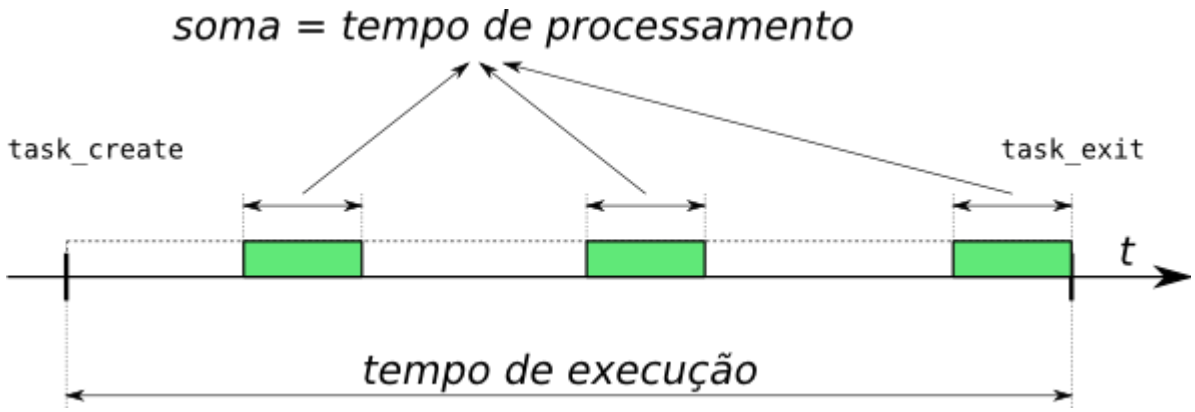
Contabilização de tarefas

Você irá adicionar mecanismos para contabilizar o uso do processador pelas tarefas em execução. Sua implementação deve produzir uma mensagem de saída com o seguinte formato, para cada tarefa que finaliza (incluindo o próprio *dispatcher*):

```
Task 17 exit: execution time 4955 ms, processor time 925 ms, 171 activations
```

Cálculo dos tempos

A figura abaixo ilustra a execução de uma determinada tarefa, de sua criação (`task_create`) ao seu encerramento (`task_exit`). As áreas em verde indicam o uso do processador. É fácil perceber como os valores de contabilização podem ser calculados:



Para a contabilização você precisará de uma **referência de tempo** (um relógio). Para isso, pode ser definida uma variável global para contar *ticks* de relógio, incrementada a cada interrupção do temporizador (1 ms). Dessa forma, essa variável indicará o número de *ticks* decorridos desde a inicialização do sistema na função `pingpong_init`, ou seja, funcionará como um relógio baseado em milissegundos.

Você deverá implementar uma função para informar às tarefas o valor corrente do relógio:

```
unsigned int systime () ;
```

Códigos de teste

- Teste 1 (saída esperada), sem usar prioridades. Neste exemplo, as tarefas devem concluir juntas, com consumo de processador e número de ativações similares.
- Teste 2 (saída esperada), usando prioridades. Neste exemplo, as tarefas devem concluir **em instantes bem distintos**, mas com consumo de processador e número de ativações similares (pois a carga computacional delas é similar).

Outras informações

- Duração estimada: 4 horas.
- Dependências:
    - Gestão de Tarefas
    - Dispatcher
    - Escalonador por prioridades (para o teste 2)
    - Preempção por Tempo
