**Henrique de S. Q. dos Santos, NUSP 10819029**

**Witor M. A. de Oliveira, NUSP 10692190**

**Gustavo T. Mastrobuono, NUSP 10734411**

**Atividade 1 - Projeto PCAM**

1. Projeto PCAM
   1. P - Particionamento - tarefa

Essa etapa define o “particionamento do problema em tarefas concorrentes”. Dada uma frase, com palavras separadas por espaços, uma tarefa será o particionamento da frase em palavras e consequente alocação de cada palavra particionada em uma linha de uma matriz. A próxima tarefa será a contagem, em paralelo, do tamanho de cada palavra que fora armazenada anteriormente. Por fim, uma última tarefa será a comparação dos tamanhos encontrados, de forma a armazenarmos o maior valor encontrado.

* 1. C - Comunicação - como cada tarefa se comunica

Essa etapa é responsável pela “definição da interação das tarefas: comunicação e sincronização”. A primeira tarefa irá retornar uma matriz contendo, em cada linha, uma palavra lida. Essa matriz será disponibilizada para o uso nas tarefas seguintes, sendo essa a única comunicação feita com a primeira tarefa. Já em relação a segunda e terceira tarefas, essas se comunicam da seguinte forma: sempre após a contagem feita pela segunda tarefa, a terceira tarefa irá receber esse valor para efetuar uma comparação com o maior valor encontrado até o momento.

* 1. A - Aglomeração - arquitetura/thread

Nessa fase, devemos “aglomerar tarefas concorrentes em processos concorrentes”. A estratégia utilizada é atribuir cada contagem do tamanho das palavras à uma *thread*, de forma que ela mantenha uma variável local (maiorLocal) contendo o tamanho da palavra lida por ela. Na tarefa seguinte (terceira), a variável maiorLocal irá ser comparada com a variável maiorGlobal, de forma a conferir qual o maior valor encontrado até o momento em todas as *threads* alocadas. Para não haver interferências, a comparação entre as variáveis maiorLocal e maiorGlobal será protegida através de um *lock*. Além disso, a variável maiorGlobal deverá ser compartilhada entre as threads, porém a maiorLocal deverá ser privada à cada uma delas.

M - Mapeamento

O mapeamento do processadores não é feito de forma explícita, cabendo ao sistema operacional decidir em qual núcleo executar as threads criadas.