

UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO

Arquivo para configuração e execução dos algoritmos.

Trabalho Final - (TF)

Programação Distribuída e Paralela – CMP 157

Raffael Bottoli Schemmer

Assume-se para este tutorial que o computador configurado possui arquitetura de máquina de 64bits (x86-64).

C:

Configurar o ambiente:

#1 - Download Ubuntu 10.04 LTS.
<http://releases.ubuntu.com/lucid/ubuntu-10.04.4-server-amd64.iso>
#2 - Em Update Manager instalar todos os Updates disponíveis.

Executar a aplicação (matrix.c):

#1 - Definir nas constantes dentro do fonte o tamanho da matriz TAM (linha 15) e o número de threads numCores (linha 21). Leia as restrições aplicadas as limitações quanto a estas entradas.
#2 - Compilando o fonte - Dispare no diretório onde o fonte estiver armazenado:
gcc -lpthread -w -O2 -o matrix matrix.c
#3 - Executando o fonte - Dispare no diretório onde o executável for criado:
./matrix

#Dica - Para visualizar resultados da multiplicação, descomente a linha 122 e execute novamente os passos #2 e #3 (Use matrizes pequenas 4/8 salvo a resolução de saída da tela).

| -- || -- || -- || -- || -- || -- || -- || -- || -- || -- || -- || -- |

Go:

Configurar o ambiente:

#1 - Download Ubuntu 10.04 LTS.
<http://releases.ubuntu.com/lucid/ubuntu-10.04.4-server-amd64.iso>
#2 - Em Update Manager instalar todos os Updates disponíveis.
#3 - Download do ambiente Go.
<http://code.google.com/p/go/downloads/detail?name=go1.2rc5.linux-amd64.tar.gz&can=2&q=>
#4 - Extrair o pacote utilizando:
tar -xvf go1.2rc5.linux-amd64.tar.gz
#5 - Mover a pasta extraída (go) para o diretório \$HOME:
mv ./go /home/userhere
#6 - Definir as seguintes variáveis de ambientes no \$PATH (Via .bashrc), executando os comandos a seguir no diretório \$HOME(/home/userhere)::
gedit .basrch
* No fim do arquivo adicione as 2 linhas abaixo:
export GOROOT=\$HOME/go
export PATH=\$PATH:\$GOROOT/bin
* Salve o arquivo.
source .bashrc

Executar a aplicação (matrix.go):

#1 - Definir nas constantes dentro do fonte o tamanho da matriz TAM (linha 23) e o número de threads numCores (linha 21). Leia as restrições aplicadas as limitações quanto a estas entradas.

#2 - Compilando e disparando o fonte - Dispare no diretório onde o fonte estiver armazenado:
go run matrix.go

#Dica - Para visualizar resultados da multiplicação, descomente a linha 172 e execute novamente o passo #2 (Use matrizes pequenas 4/8 salvo a resolução de saída da tela).

| -- || -- || -- || -- || -- || -- || -- || -- || -- || -- || -- ||

Cilk:

Configurar o ambiente:

#1 - Download Ubuntu 10.04 LTS.
<http://releases.ubuntu.com/lucid/ubuntu-10.04.4-server-amd64.iso>
#2 - Em Update Manager instalar todos os Updates disponíveis.
#3 - Download do SDK Cilk (Site da Intel):
<https://secure-software.intel.com/en-us/system/files/article/164673/cilk-8503-x86-64.release.tar.gz>
#4 - Extrair o pacote utilizando:
tar -xvf cilk-8503-x86-64.release.tar.gz
#5 - Mover a pasta extraída (cilk) para o diretório \$HOME:
mv ./cilk /home/userhere
#6 - Definir as seguintes variáveis de ambientes no \$PATH (Via .bashrc), executando os comandos a seguir no diretório \$HOME(/home/userhere):
gedit .basrch
* No fim do arquivo adicione as 2 linhas abaixo:
export CILK=/home/usercilk/cilk
export PATH=\$PATH:\$CILK/bin
* Salve o arquivo.
source .bashrc

Executar a aplicação (matrix.cpp):

#1 - Definir nas constantes dentro do fonte o tamanho da matriz TAM (linha 13). Leia as restrições aplicadas as limitações quanto a estas entradas.
#2 - Compilando o fonte - Dispare no diretório onde o fonte estiver armazenado:
cilk++ -o matrix matrix.cpp
#3 - Executando o fonte - Dispare no diretório onde o fonte estiver armazenado, juntamente com o parâmetro (-cilk_set_worker_count 4) que define o número de threads (neste caso 4) que a estrutura cilk_for deverá executar. Leia as restrições disponíveis no código fonte aplicadas as limitações quanto ao número de threads (No caso, somente variações de base 2 (4/8/16/32) serão aceitas para o número de threads.):
./matrix -cilk_set_worker_count 4

#Dica - Para visualizar resultados da multiplicação, descomente a linha 88 e execute novamente os passos #2 e #3 (Use matrizes pequenas 4/8 salvo a resolução de saída da tela).

| -- || -- || -- || -- || -- || -- || -- || -- || -- || -- || -- ||

UPC:

Configurar o ambiente:

```

#1 - Download Ubuntu 12.04 LTS.
http://mirror.pop-sc.rnp.br/mirror/ubuntu//precise/ubuntu-12.04.3-
desktop-amd64.iso
#2 - Em Update Manager instalar todos os Updates disponíveis.
#3 - Download do compilador GNU UPC:
http://www.gccupc.org/downloads/upc/rls/upc-4.9.0.1/upc-4.9.0.1-x86_64-
linux-ubuntu12.4.tar.gz
#4 - Extrair o pacote utilizando:
tar -xvf upc-4.9.0.1-x86_64-linux-ubuntu12.4.tar.gz
#5 - Mover a pasta extraída (cilk) para o diretório $HOME:
mv ./usr /home/userhere
#6 - Definir as seguintes variáveis de ambientes no $PATH (Via .bashrc),
executando os comandos a seguir no diretório $HOME(/home/userhere):
gedit .basrch
* No fim do arquivo adicione as 5 linhas abaixo:
LIBRARY_PATH=/usr/lib/x86_64-linux-gnu/
C_INCLUDE_PATH=/usr/include/x86_64-linux-gnu
CPLUS_INCLUDE_PATH=/usr/include/x86_64-linux-gnu
export UPC=/home/userhere/usr/local/gupc
export PATH=$PATH:$UPC/bin
* Salve o arquivo.
source .bashrc
#7 - Instalação da biblioteca lnuma (Necessária para compilação dos
fontes):
sudo su apt-get install libnuma-dev

```

Executar a aplicação (matrix.upc):

```

#1 - Definir nas constantes dentro do fonte o tamanho da matriz TAM
(linha 14). Leia as restrições aplicadas as limitações quanto a estas
entradas.
#2 - Compilando o fonte - Dispare no diretório onde o fonte estiver
armazenado, juntamente com o parâmetro (-fupc-threads-4) que define o
número de threads (ou réplicas da função main e suas sub funções, neste
caso 4) que o compilador UPC deverá gerar da aplicação:
gupc -fupc-threads-2 -o matrix matrix.upc
#3 - Executando o fonte - Dispare no diretório onde o fonte estiver
armazenado:
./matrix

```

#Dica - Para visualizar resultados da multiplicação, descomente a linha 97 e execute novamente os passos #2 e #3 (Use matrizes pequenas 4/8 salvo a resolução de saída da tela).

```
| -- || -- || -- || -- || -- || -- || -- || -- || -- || -- ||
```