DCC
DEPARTAMENTO DE
CIÊNCIA DA COMPUTAÇÃO

UFMG
UNIVERSIDADE FEDERAL
DE MINAS GERAIS

Estruturas de Dados

Aula Prática 2

Professores: Gisele L Pappa
Wagner Meira Jr

Objetivos

- Depurar código fonte
  - gdb.
- Instrumentar aplicações para coletar dados de desempenho e localidade de referência
  - memlog
- Planejar e realizar experimentos.
  - Desempenho computacional
  - Localidade de referência
- Analisar resultados experimentais
  - analisamem
  - gnuplot
- Depurar desempenho
  - gprof

Preparação

- Assistir vídeos do Material Adicional
  - Localidade de Referência
  - Tipos Abstratos de Dados
  - Tipo Abstrato de Dados Matriz
  - Análise de Desempenho TAD Matriz
  - Análise de Localidade de Referência do TAD Matriz
  - Depuração de Desempenho do TAD Matriz

gdb - Depurar código fonte

- Cenário: erro de lógica no malloc do TAD vetor.

1. Execução gera falha de segmentação
2. Compila com -g
3. Executa no gdb
4. Localiza erro
5. E se no gdb o erro sumir?

- Outros recursos:
  - valgrind
  - electric fence

Instrumentar com memlog

```
int iniciaMemLog(char * nome);
int ativaMemLog();
int desativaMemLog();
int defineFaseMemLog(int f);
int leMemLog(long int pos, long int tam, int id);
int escreveMemLog(long int pos, long int tam, int id);
int finalizaMemLog();
```

```
// macros para maior eficiencia
#define LEMEMLOG(pos,tam,id)
((void) ((ml.ativo==MLATIVO)?leMemLog(pos,tam,id):0))
#define ESCREVEMEMLOG(pos,tam,id)
((void) ((ml.ativo==MLATIVO)?escreveMemLog(pos,tam,id):0))
```

Planejar e executar experimentos

- Desempenho computacional
  - Selecionar dimensões
  - Evitar tempos de execução pequenos (<100ms)
  - Não registrar acessos a memória
- Localidade de referência
  - Selecionar dimensões
  - Registrar acessos à memória
    - Custo de monitoramento
    - Facilidade de visualização

Analisar resultados experimentais

- Desempenho computacional
  - Tabelas
  - Gráficos (gnuplot)
  - Contraste com a análise de complexidade
- Localidade de referência
  - Aplicativo analisamem
  - Gráficos
    - Mapa de acesso
    - Distância de pilha

Depurar desempenho

- Ferramenta gprof
  - Utilizar gprof com as medidas de desempenho computacional e análise de complexidade
    - Análise isolada por função
    - Análise cumulativa
  - Ferramentas são baseadas em amostragem
    - Trabalhar com dimensões significativas
  - Outros exemplos:
    - http://euccas.github.io/blog/20170827/cpu-profiling-tools-on-linux.html

O que fazer:

1. Obter aulapratica02.zip do minha.ufmg
2. Inspecionar o código: vetop.c e vet.c
3. Revisar dimensões utilizadas
4. Executar os vários experimentos (make)
5. Criar um documento contendo
   a. Resultados computacionais
   b. Análise de localidade de referência
      i. Mapas de acesso
      ii. Distância de pilha
   c. Depuração desempenho
   d. Saída gprof
6. Submeter zip com todos os arquivos gerados