Relatório EP2 MAC0438

Ciro S. Costa, 8071488

May, 25, 2015

Sobre

Configuração das máquinas de teste

Feature	Máquina1
CPU	i5-460M
Cores	2
Threads	4
Cache	3M
Geração	1st
Freq	2.53GHz
RAM	3072 MB
Storage	HDD

Compilador: Clang++ Versão 3.5

Benchmark

Para conseguir analisar os resultados obtidos com a implementação utilizou-se de benchmarking através de repetida execução de testes automatizados.

Busca-se o mensurar tempo de execução através da biblioteca padrão (C++11) com a implementação de um pequeno conjunto de macros utilizadas para capturar os resultados obtidos das repetidas execuções.

Com os dados coletados, busca-se então validar as seguintes hipóteses:

1. Utilizando o modo 'f' teremos menos eficiência do que com o método 'm' dado que o segundo, apesar de precisar parar na barreira tanto quanto o primeiro, precisa realizar menos etapas de redução.

2. Maior eficiência para número de threads utilizadas igual ao número suportado pela máquina (especificando 0 no parâmetro de entrada, por tanto).

Resultado

Não consegui concluir o EP a ponto de mensurar como esperava, além de gerar o relatório de benchmark de maneira adequada. Tive sucesso em implementar o modo 'm' de execução do programa. Faltou, também, atenção à suit de testes, além de melhor controle da precisão do GMP (está sendo sempre configurado para um número muito grande).

Barreiras e Biblioteca de Múltipla Precisão

A sincronização do cálculo das threads (etapas de redução e verificação) contou com a implementação de uma barreira de duas fases e semáforos com a possibilidade de 'pre-carregar' o semáforo com mais de um sinal. Isto permitiu uma representação simples do conceito.

Para que fosse possível efetuar cálculos com tamanha precisão foi utilizada a biblioteca GMP (Gnu Multiple Precision Arithmetic Library) que nos permite utilizar de precisão artitrária (limitada à memória disponível). A ideia básica é tratar grandes números como uma composição de pequenas partes e aplicar regras básicas de aritimética parte a parte.

Para cada tipo que a biblioteca define (inteiros, floats e racionais), uma representação interna diferente é utilizada. No caso do Mpf (multiple precision float) - o mais utilizado na implementação - a estrutura é dada por duas partes: o conjunto referente ao expoente que trata de determinar o ponto de raíz e a segunda para a determinação do significando, através da utilização de simples doubles para sua representação.