Universidade de São Paulo

Instituto de Ciências Matemáticas e de Computação

Desenvolvimento de Código Otimizado 2023.2

Atividade 3: Flags de otimização

Gabriel da Cunha Dertoni	11795717
Matheus Ventura de Sousa	11345541
Pedro Lucas de Moliner de Castro	11795784
Vitor Caetano Brustolin	11795589

Profa. Dr. Sarita Mazzini Bruschi

São Carlos, Outubro de 2023

1 Introdução

O objetivo dessa atividade é comparar os efeitos das flags de otimização $-O0$, $-O1$, $-O2$, $-O3$ e $-Os$ no tempo de execução, compilação e tamanho do binário final. Para isso, foram selecionados dois exemplos do CLBG, o fasta e o binarytrees.

2 Ambiente de execução

O experimento foi compilado usando `gcc` e executado em uma máquina com as especificações descritas na tabela a seguir:

Nome do modelo	Intel(R) Core(TM) i5-9400F CPU @ 2.90GHz
Arquitetura	x86_64
Modos operacionais da CPU	32-bit, 64-bit
Ordem dos bytes	Little Endian
CPUs	6
Threads por núcleo	1
Núcleos por soquete	6
Soquetes	1
Núcleos de NUMA	1
Frequência máxima	4100 MHz
Frequência mínima	800 MHz
Cache de L1d	192 KiB (6 instâncias)
Cache de L1i	192 KiB (6 instâncias)
Cache de L2	1.5 MiB (6 instâncias)
Cache de L3	9 MiB (1 instância)

3 Scripts auxiliares

Para facilitar a execução dos testes e calcular as médias, foi desenvolvido um *script python* que pode ser executado passando como argumento o nome do algoritmo que deve ser executado.

```
python3 scripts/benchmark.py fasta
python3 scripts/benchmark.py binarytrees
```

4 Análise dos tempos

4.1 Fasta

Fasta em $-O0$ com tamanho 16416 bytes

Tempo de compilação (ms)	Tempos de execução (ms)
0.036671	0.000808
0.036706	0.000811
0.036942	0.000816
0.036629	0.000819
0.036704	0.000810
0.036955	0.000915
0.036706	0.000806
0.036587	0.000810
0.036846	0.000816
0.036876	0.000816
0.036838	0.000807
0.036744	0.000802
0.036670	0.000805
0.036769	0.000809
0.036541	0.000803
0.036746 ± 0.000069	0.000817 ± 0.000015

Fasta em $-O1$ com tamanho 16296 bytes

Tempo de compilação (ms)	Tempos de execução (ms)
0.048139	0.000678
0.047736	0.000665
0.047793	0.000682
0.047807	0.000684
0.047842	0.000675
0.047825	0.000670
0.047571	0.000714
0.047978	0.000676
0.047814	0.000673
0.047657	0.000680
0.047499	0.000684
0.047794	0.000692
0.047585	0.000675
0.047639	0.000680
0.047715	0.000670
0.047760 ± 0.000090	0.000680 ± 0.000006

Fasta em $-O2$ com tamanho 16296 bytes

Tempo de compilação (ms)	Tempos de execução (ms)
0.054395	0.000661
0.054160	0.000658
0.054171	0.000663
0.053901	0.000745
0.054159	0.000671
0.053970	0.000657
0.054022	0.000655
0.054098	0.000665
0.054180	0.000702
0.053898	0.000654
0.054016	0.000664
0.054014	0.000663
0.053964	0.000665
0.053778	0.000656
0.053850	0.000668
0.054038 ± 0.000088	0.000670 ± 0.000013

Fasta em $-O3$ com tamanho 16296 bytes

Tempo de compilação (ms)	Tempos de execução (ms)
0.054943	0.000664
0.054461	0.000656
0.054468	0.000655
0.054382	0.000653
0.054675	0.000662
0.054537	0.000665
0.054463	0.000662
0.054597	0.000651
0.054429	0.000693
0.054493	0.000664
0.054745	0.000667
0.054402	0.000658
0.054329	0.000654
0.054586	0.000656
0.054438	0.000671
0.054530 ± 0.000088	0.000662 ± 0.000006

Fasta em $-Os$ com tamanha 16256 bytes

Tempo de compilação (ms)	Tempos de execução (ms)
0.048716	0.000835
0.048422	0.000779
0.048553	0.000791
0.048410	0.000791
0.048385	0.000788
0.048690	0.000788
0.048351	0.000790
0.048636	0.000823
0.048349	0.000793
0.048443	0.000788
0.048264	0.000788
0.048423	0.000790
0.048234	0.000838
0.048171	0.000784
0.048748	0.000832
0.048453 ± 0.000099	0.000800 ± 0.000011

Para esse algoritmo, nota-se um aumento significativo no tempo de compilação de $-O0$ para $-O1$ ou $-Os$ e de $-O1$ para $-O2$ ou $-O3$. No quesito tempo de execução, $-O0$ e $-Os$ são similares e há uma queda grande quando usando $-O1$, $-O2$ e $-O3$, entre esses três também há uma diminuição de tempo mas não tão grande. O tamanho do binário final foi o mesmo independente da flag.

4.2 Binarytrees

Binarytrees em $-O0$ com tamanho 16504 bytes

Tempo de compilação (ms)	**Tempos de execução (ms)**
0.038172	0.005742
0.037804	0.005761
0.038007	0.005719
0.038071	0.005734
0.037699	0.005852
0.037619	0.005792
0.037670	0.005757
0.038000	0.005806
0.037945	0.005777
0.038039	0.005884
0.038102	0.006055
0.037868	0.005808
0.038049	0.005847
0.038053	0.005819
0.038083	0.005812
0.037945 ± 0.000096	0.005811 ± 0.000045

Binarytrees em $-O1$ com tamanho 16480 bytes

Tempo de compilação (ms)	Tempos de execução (ms)
0.048934	0.005250
0.048288	0.005153
0.048096	0.005205
0.048209	0.004967
0.048195	0.005001
0.048318	0.005273
0.048123	0.005115
0.048190	0.005320
0.048289	0.005124
0.047980	0.005083
0.048431	0.005035
0.048431	0.005071
0.048123	0.005084
0.048302	0.005167
0.048154	0.005278
0.048271 ± 0.000122	0.005142 ± 0.000059

Binarytrees em $-O2$ com tamanho 16480 bytes

Tempo de compilação (ms)	Tempos de execução (ms)
0.084015	0.004928
0.084232	0.004939
0.084210	0.005094
0.084071	0.004973
0.084439	0.004880
0.084390	0.005165
0.084300	0.005061
0.083954	0.004914
0.084232	0.005022
0.084531	0.004979
0.084070	0.004918
0.084664	0.004909
0.083977	0.004972
0.083753	0.004922
0.084060	0.005004
0.084193 ± 0.000135	0.004979 ± 0.000044

Binarytrees em $-O3$ com tamanho 20576 bytes

Tempo de compilação (ms)	Tempos de execução (ms)
0.146292	0.004895
0.144952	0.004974
0.145025	0.004903
0.145326	0.005098
0.146501	0.005084
0.146416	0.004983
0.145388	0.005051
0.145209	0.005180
0.145745	0.004902
0.145157	0.004931
0.145405	0.004996
0.145287	0.005022
0.144220	0.004973
0.147626	0.005024
0.147133	0.005001
0.145712 ± 0.000502	0.005001 ± 0.000044

Binarytrees em $-Os$ com tamanho 16480 bytes

Tempo de compilação (ms)	Tempos de execução (ms)
0.054113	0.005323
0.054483	0.005121
0.054032	0.005075
0.054310	0.005080
0.054229	0.005114
0.053896	0.005013
0.053560	0.004982
0.053317	0.005106
0.052821	0.005004
0.053001	0.005086
0.054838	0.005131
0.055142	0.005169
0.055055	0.005032
0.054717	0.005056
0.053984	0.005030
0.054100 ± 0.000390	0.005088 ± 0.000046

Nesse caso de teste, nota-se que a diferença no tempo de compilação é sempre grande, sendo do menor para o maior $-O0$, $-O1$, $-Os$, $-O2$ e, finalmente, $-O3$ que tomou quase 4 vezes o tempo de $-O0$. A disparidade no tempo de execução, em contrapartida, não foi tão grande, diminuindo de 0.0058 ms em $-O0$ para em torno de 0.0050 ms com otimizações ligadas. Finalmente, aqui houve diferença no tamanho do binário, sendo que o maior foi o $-O3$, o segundo maior $-O0$ e o menor as outras três empatadas. Possivelmente essa diminuição de tamanho foi consequência de algum *inline* no código, já que $-O1$ e $-O2$ obtiveram o mesmo resultado de $-Os$.

5 Conclusão

Em conclusão, nota-se uma correlação clara entre a ativação de flags de otimização de tempo ou tamanho e o aumento no tempo de compilação. Além disso, as flags de otimização de tempo $-O1$, $-O2$ e $-O3$ se mostraram efetivas, mas o ganho de performance diminuiu a cada nível. Por fim, a otimização de tamanho $-Os$ não se mostrou muito eficiente, pelo menos em códigos pequenos, criando binários do mesmo tamanho que ao menos duas outras opções de otimização em ambos exemplos.