Bilgisayar Organizasyonu Ödev - 4

Öğrenci Adı: Burak Atalay

Öğrenci Numarası: 22011641

Dersin Eğitmeni: Ali Can Karaca

1- CPU Bilgileri:

Cihaz özellikleri Kopyala

Cihaz adı	Costanzas
İşlemci	Intel(R) Core(TM) i5-10300H CPU @ 2.50GHz 2.50 GHz
Takılı RAM	8,00 GB (kullanılabilir: 7,87 GB)
Cihaz Kimliği	B8460CE8-2E63-4547-A422-D599FA616399
Ürün Kimliği	00325-82118-27735-AAOEM
Sistem türü	64 bit işletim sistemi, x64 tabanlı işlemci
Kalem ve dokunma	Bu görüntü biriminde kalem girdisi veya dokunarak giriş yok

Architecture / Microarchitecture

Instruction set	x86
Microarchitecture	Comet Lake
Processor core	Comet Lake-H
Core stepping	R1
CPUIDs	A0652
Manufacturing process	0.014 micron
Data width	64 bit
The number of CPU cores	**4**
The number of threads	**8**
Floating Point Unit	Integrated
Level 1 cache size	4 x 32 KB 8-way set associative instruction caches 4 x 32 KB 8-way set associative data caches
Level 2 cache size	4 x 256 KB 4-way set associative caches
Level 3 cache size	**8 MB 16-way set associative shared cache**
Physical memory	128 GB
Multiprocessing	Uniprocessor
Extensions & Technologies	▪ MMX instructions ▪ SSE / Streaming SIMD Extensions ▪ SSE2 / Streaming SIMD Extensions 2 ▪ SSE3 / Streaming SIMD Extensions 3 ▪ SSSE3 / Supplemental Streaming SIMD Extensions 3 ▪ SSE4 / SSE4.1 + SSE4.2 / Streaming SIMD Extensions 4 ▪ AES / Advanced Encryption Standard instructions ▪ AVX / Advanced Vector Extensions ▪ AVX2 / Advanced Vector Extensions 2.0 ▪ BMI / BMI1 + BMI2 / Bit Manipulation instructions ▪ F16C / 16-bit Floating-Point conversion instructions ▪ FMA3 / 3-operand Fused Multiply-Add instructions ▪ EM64T / Extended Memory 64 technology / Intel 64 ▪ HT / Hyper-Threading technology ▪ VT-x / Virtualization technology ▪ VT-d / Virtualization for directed I/O ▪ TBT 2.0 / Turbo Boost technology 2.0 ▪ TSX / Transactional Synchronization Extensions
Security Features	▪ NX / XD / Execute disable bit ▪ MPX / Memory Protection Extensions ▪ SGX / Software Guard Extensions [1]
Low power features	Enhanced SpeedStep technology

2- Teorik Analiz:

Birinci ve üçüncü kodlarda, ikinci kodun aksine B matrisinde satırı değil sütunu geziyoruz. Çünkü diğer ikisinde en iç döngüde k var ve bu yüzden B matrisinin satırını değil de sütünun gezmiş oluyoruz. Bu ise miss oranını artırıyor. Çünkü kodda bir adrese erişilmek istendiğinde, bu adresteki değer ve sonraki cache line'a sığacak kadar değer cache line'a alınıp buradan cache'e yazılır. Cache'deki değer değil de yeniden memory'ye gidilip cache line'a değer alınırsa bu miss oranını artırır, hit oranı ve miss oranı arasında ters orantı vardır. Mesela birinci ve üçüncü kodda B matrisinin bir sütunundaki ilk değerini sorguladık, cache line büyük ihtimalle sütunun ikinci elemanını alamayacak. Bu sebeple iç döngünün ikinci iterasyonunda cache'ten okuma yapamayıp tekrar memory'e soracak ve cache line'a elemanları alacak. Sonuç olarak ikinci kodun miss oranı diğer iki koddakine oranla düşük, hit oranı ise yüksektir. Bu da daha iyi bir cache performansı sunar. İkinci kod, birinci ve üçüncüye göre daha hızlı çalışır.

Dimension'ın 256 değil de 64 olması ise matrislerin cache'e daha rahat sığmasına ve memory'e daha az sorgu yapılmasını sağlar. Bu sebeple kodların hit oranları yükselir, miss oranları düşer.

3- Valgrid Sonuçları:

Dimension = 256

1.c

```
costanza@Costanzas:/mnt/c/hw4$ valgrind --tool=cachegrind --cache-sim=yes ./1_256.out
==23430== Cachegrind, a cache and branch-prediction profiler
==23430== Copyright (C) 2002-2017, and GNU GPL'd, by Nicholas Nethercote et al.
==23430== Using Valgrind-3.18.1 and LibVEX; rerun with -h for copyright info
==23430== Command: ./1_256.out
==23430==
--23430-- warning: L3 cache found, using its data for the LL simulation.

secs:4.262208
==23430==
==23430== I   refs:      783,291,123
==23430== I1  misses:          1,455
==23430== LLi misses:          1,429
==23430== I1  miss rate:        0.00%
==23430== LLi miss rate:        0.00%
==23430==
==23430== D   refs:      306,629,187  (288,657,650 rd   + 17,971,537 wr)
==23430== D1  misses:     16,894,739  ( 16,869,502 rd   +     25,237 wr)
==23430== LLd misses:         26,486  (      1,314 rd   +     25,172 wr)
==23430== D1  miss rate:         5.5% (        5.8%     +        0.1%  )
==23430== LLd miss rate:         0.0% (        0.0%     +        0.1%  )
==23430==
==23430== LL refs:        16,896,194  ( 16,870,957 rd   +     25,237 wr)
==23430== LL misses:          27,915  (      2,743 rd   +     25,172 wr)
==23430== LL miss rate:          0.0% (        0.0%     +        0.1%  )
```

2.c

```
costanza@Costanzas:/mnt/c/hw4$ valgrind --tool=cachegrind --cache-sim=yes ./2_256.out
==23315== Cachegrind, a cache and branch-prediction profiler
==23315== Copyright (C) 2002-2017, and GNU GPL'd, by Nicholas Nethercote et al.
==23315== Using Valgrind-3.18.1 and LibVEX; rerun with -h for copyright info
==23315== Command: ./2_256.out
==23315==
--23315-- warning: L3 cache found, using its data for the LL simulation.

secs:3.420467
==23315==
==23315== I   refs:      783,291,105
==23315== I1  misses:          1,451
==23315== LLi misses:          1,426
==23315== I1  miss rate:        0.00%
==23315== LLi miss rate:        0.00%
==23315==
==23315== D   refs:      306,629,181  (288,657,645 rd   + 17,971,536 wr)
==23315== D1  misses:      2,140,691  (  2,115,454 rd   +     25,237 wr)
==23315== LLd misses:         26,486  (      1,314 rd   +     25,172 wr)
==23315== D1  miss rate:         0.7% (        0.7%     +        0.1%  )
==23315== LLd miss rate:         0.0% (        0.0%     +        0.1%  )
==23315==
==23315== LL refs:         2,142,142  (  2,116,905 rd   +     25,237 wr)
==23315== LL misses:          27,912  (      2,740 rd   +     25,172 wr)
==23315== LL miss rate:          0.0% (        0.0%     +        0.1%  )
costanza@Costanzas:/mnt/c/hw4$ valgrind --tool=cachegrind --cache-sim=yes ./3_256.out
```

3.c

```
costanza@Costanzas:/mnt/c/hw4$ valgrind --tool=cachegrind --cache-sim=yes ./3_256.out
==23352== Cachegrind, a cache and branch-prediction profiler
==23352== Copyright (C) 2002-2017, and GNU GPL'd, by Nicholas Nethercote et al.
==23352== Using Valgrind-3.18.1 and LibVEX; rerun with -h for copyright info
==23352== Command: ./3_256.out
==23352==
--23352-- warning: L3 cache found, using its data for the LL simulation.

secs:4.062655
==23352==
==23352== I   refs:      783,291,123
==23352== I1  misses:          1,455
==23352== LLi misses:          1,429
==23352== I1  miss rate:        0.00%
==23352== LLi miss rate:        0.00%
==23352==
==23352== D   refs:      306,629,187  (288,657,650 rd   + 17,971,537 wr)
==23352== D1  misses:     18,967,057  ( 18,941,820 rd   +     25,237 wr)
==23352== LLd misses:         26,486  (      1,314 rd   +     25,172 wr)
==23352== D1  miss rate:         6.2% (        6.6%     +        0.1%  )
==23352== LLd miss rate:         0.0% (        0.0%     +        0.1%  )
==23352==
==23352== LL refs:        18,968,512  ( 18,943,275 rd   +     25,237 wr)
==23352== LL misses:          27,915  (      2,743 rd   +     25,172 wr)
==23352== LL miss rate:          0.0% (        0.0%     +        0.1%  )
costanza@Costanzas:/mnt/c/hw4$ valgrind --tool=cachegrind --cache-sim=yes ./1_256.out
```

Dimension = 64

1.c

```
costanza@Costanzas:/mnt/c/hw4$ valgrind --tool=cachegrind --cache-sim=yes ./1.out
==25340== Cachegrind, a cache and branch-prediction profiler
==25340== Copyright (C) 2002-2017, and GNU GPL'd, by Nicholas Nethercote et al.
==25340== Using Valgrind-3.18.1 and LibVEX; rerun with -h for copyright info
==25340== Command: ./1.out
==25340==
--25340-- warning: L3 cache found, using its data for the LL simulation.

secs:0.032086
==25340==
==25340== I   refs:      12,919,852
==25340== I1  misses:         1,479
==25340== LLi misses:         1,453
==25340== I1  miss rate:       0.01%
==25340== LLi miss rate:       0.01%
==25340==
==25340== D   refs:       5,055,604  (4,705,425 rd   + 350,179 wr)
==25340== D1  misses:        52,697  (   50,499 rd   +   2,198 wr)
==25340== LLd misses:         3,449  (    1,319 rd   +   2,130 wr)
==25340== D1  miss rate:        1.0% (      1.1%     +     0.6%  )
==25340== LLd miss rate:        0.1% (      0.0%     +     0.6%  )
==25340==
==25340== LL refs:           54,176  (   51,978 rd   +   2,198 wr)
==25340== LL misses:          4,902  (    2,772 rd   +   2,130 wr)
==25340== LL miss rate:         0.0% (      0.0%     +     0.6%  )
```

2.c

```
costanza@Costanzas:/mnt/c/hw4$ valgrind --tool=cachegrind --cache-sim=yes ./2.out
==25246== Cachegrind, a cache and branch-prediction profiler
==25246== Copyright (C) 2002-2017, and GNU GPL'd, by Nicholas Nethercote et al.
==25246== Using Valgrind-3.18.1 and LibVEX; rerun with -h for copyright info
==25246== Command: ./2.out
==25246==
--25246-- warning: L3 cache found, using its data for the LL simulation.

secs:0.031665
==25246==
==25246== I   refs:      12,919,863
==25246== I1  misses:         1,479
==25246== LLi misses:         1,453
==25246== I1  miss rate:       0.01%
==25246== LLi miss rate:       0.01%
==25246==
==25246== D   refs:       5,055,608  (4,705,429 rd   + 350,179 wr)
==25246== D1  misses:        14,258  (   12,060 rd   +   2,198 wr)
==25246== LLd misses:         3,449  (    1,319 rd   +   2,130 wr)
==25246== D1  miss rate:        0.3% (      0.3%     +     0.6%  )
==25246== LLd miss rate:        0.1% (      0.0%     +     0.6%  )
==25246==
==25246== LL refs:           15,737  (   13,539 rd   +   2,198 wr)
==25246== LL misses:          4,902  (    2,772 rd   +   2,130 wr)
==25246== LL miss rate:         0.0% (      0.0%     +     0.6%  )
```

3.c

```
costanza@Costanzas:/mnt/c/hw4$ valgrind --tool=cachegrind --cache-sim=yes ./3.out
==25256== Cachegrind, a cache and branch-prediction profiler
==25256== Copyright (C) 2002-2017, and GNU GPL'd, by Nicholas Nethercote et al.
==25256== Using Valgrind-3.18.1 and LibVEX; rerun with -h for copyright info
==25256== Command: ./3.out
==25256==
--25256-- warning: L3 cache found, using its data for the LL simulation.

secs:0.033221
==25256==
==25256== I   refs:      12,919,867
==25256== I1  misses:         1,479
==25256== LLi misses:         1,453
==25256== I1  miss rate:       0.01%
==25256== LLi miss rate:       0.01%
==25256==
==25256== D   refs:       5,055,609  (4,705,430 rd   + 350,179 wr)
==25256== D1  misses:        76,861  (   74,663 rd   +   2,198 wr)
==25256== LLd misses:         3,449  (    1,319 rd   +   2,130 wr)
==25256== D1  miss rate:        1.5% (      1.6%     +     0.6%  )
==25256== LLd miss rate:        0.1% (      0.0%     +     0.6%  )
==25256==
==25256== LL refs:           78,340  (   76,142 rd   +   2,198 wr)
==25256== LL misses:          4,902  (    2,772 rd   +   2,130 wr)
==25256== LL miss rate:         0.0% (      0.0%     +     0.6%  )
```

4- Sonuç:

Sonuçlardan da görüleceği üzere teorik analizde tahmin edildiği gibi ikinci kod özellikle dimension 256 iken açık ara daha iyi durumda. Birinci ve üçüncü kodlar ise benzer performanslara sahip.