CSO Assignment-2

Cache Optimisation

Made by -

Rishabh Khanna (2019113025)

Task-0 [System Details] :

System's operating system :

-Version-
Kernel : Linux 5.4.0-72-generic (x86_64)
Version : #80~18.04.1-Ubuntu SMP Mon Apr 12 23:26:25 UTC 2021
C Library : GNU C Library / (Ubuntu GLIBC 2.27-3ubuntu1.2) 2.27
Distribution : Ubuntu 18.04.5 LTS
-Current Session-
Computer Name : rishabh-predator
User Name : rishabh_ (Rishabh Khanna)
Language : en_IN (en_IN:en)
Home Directory : /home/rishabh_
-Misc-
Uptime : 1 hour 52 minutes
Load Average : 2.39, 1.98, 1.99
Available entropy in /dev/random : 3856 bits (healthy)

Kernel modules :

-Loaded Modules-
rfcomm : Bluetooth RFCOMM ver 1.11
xfrm_user
xfrm4_tunnel
tunnel4
l2tp_ppp : PPP over L2TP over UDP
ipcomp : IP Payload Compression Protocol (IPComp/IPv4) - RFC3173
l2tp_netlink : L2TP netlink
xfrm_ipcomp : IP Payload Compression Protocol (IPComp) - RFC3173
esp4
l2tp_core : L2TP core
ah4
ip6_udp_tunnel
af_key
udp_tunnel
xfrm_algo
pppox : PPP over Ethernet driver (generic socket layer)
ccm : Counter with CBC MAC
cmac : CMAC keyed hash algorithm
bnep : Bluetooth BNEP ver 1.3
nls_iso8859_1
8250_dw : Synopsys DesignWare 8250 serial port driver
mei_hdcp : MEI HDCP
joydev : Joystick device interfaces
nvidia_uvm
hid_multitouch : HID multitouch panels
snd_sof_pci
intel_rapl_msr : Driver for Intel RAPL (Running Average Power Limit) control via MSR interface
hid_generic : HID generic driver
intel_rapl_common : Intel Runtime Average Power Limit (RAPL) common code
snd_sof_intel_hda_common
snd_soc_hdac_hda : ASoC Extensions for legacy HDA Drivers
uvcvideo : USB Video Class driver
snd_sof_intel_hda
snd_sof_intel_byt
x86_pkg_temp_thermal : X86 PKG TEMP Thermal Driver
snd_sof_intel_ipc
nvidia_drm
snd_sof : Sound Open Firmware (SOF) Core
snd_sof_xtensa_dsp : SOF Xtensa DSP support
intel_powerclamp : Package Level C-state Idle Injection for Intel CPUs
nvidia_modeset
videobuf2_vmalloc : vmalloc memory handling routines for videobuf2
coretemp : Intel Core temperature monitor
snd_hda_ext_core : HDA extended core
videobuf2_memops : common memory handling routines for videobuf2
videobuf2_v4l2 : Driver helper framework for Video for Linux 2

snd_soc_acpi_intel_match : Intel Common ACPI Match module
kvm_intel
videobuf2_common : Media buffer core framework
videodev : Video4Linux2 core driver
nvidia
snd_soc_acpi : ALSA SoC ACPI module
kvm
mc : Device node registration for media drivers
snd_soc_core : ALSA SoC Core
snd_compress : ALSA Compressed offload framework
iwlmvm : The new Intel(R) wireless AGN driver for Linux
ac97_bus
btusb : Generic Bluetooth USB driver ver 0.8
crct10dif_pclmul : T10 DIF CRC calculation accelerated with PCLMULQDQ.
crc32_pclmul
ghash_clmulni_intel : GHASH hash function, accelerated by PCLMULQDQ-NI
snd_pcm_dmaengine
btrtl : Bluetooth support for Realtek devices ver 0.1
aesni_intel : Rijndael (AES) Cipher Algorithm, Intel AES-NI instructions optimized
snd_hda_codec_realtek : Realtek HD-audio codec
btbcm : Bluetooth support for Broadcom devices ver 0.1
btintel : Bluetooth support for Intel devices ver 0.1
mac80211 : IEEE 802.11 subsystem
crypto_simd
snd_hda_codec_generic : Generic HD-audio codec parser
cryptd : Software async crypto daemon
libarc4
bluetooth : Bluetooth Core ver 2.22
ledtrig_audio : LED trigger for audio mute control
glue_helper
ecdh_generic : ECDH generic algorithm
ecc
rapl
snd_seq_midi : Advanced Linux Sound Architecture sequencer MIDI synth.
snd_hda_codec_hdmi : HDMI HD-audio codec
input_leds : Input -> LEDs Bridge
snd_seq_midi_event : MIDI byte <-> sequencer event coder
intel_cstate
serio_raw : Raw serio driver
snd_rawmidi : Midlevel RawMidi code for ALSA.
i915 : Intel Graphics
rtsx_pci_ms : Realtek PCI-E Memstick Card Host Driver
snd_seq : Advanced Linux Sound Architecture sequencer.
snd_hda_intel : Intel HDA driver
wmi_bmof : WMI embedded Binary MOF driver
snd_intel_dspcfg : Intel DSP config driver
acer_wmi : Acer Laptop WMI Extras Driver
sparse_keymap : Generic support for sparse keymaps
intel_wmi_thunderbolt : Intel WMI Thunderbolt force power driver
snd_hda_codec : HDA codec core
snd_hda_core : HD-audio bus
memstick : Sony MemoryStick core driver
snd_hwdep : Hardware dependent layer
drm_kms_helper : DRM KMS helper
snd_pcm : Midlevel PCM code for ALSA.
drm : DRM shared core routines
mei_me : Intel(R) Management Engine Interface
mei : Intel(R) Management Engine Interface
iwlwifi : Intel(R) Wireless WiFi driver for Linux
intel_lpss_pci : Intel LPSS PCI driver
intel_lpss : Intel LPSS core driver
i2c_algo_bit : I2C-Bus bit-banging algorithm
fb_sys_fops : Generic file read (fb in system RAM)
syscopyarea : Generic copyarea (sys-to-sys)
sysfillrect : Generic fill rectangle (sys-to-sys)
sysimgblt : 1-bit/8-bit to 1-32 bit color expansion (sys-to-sys)
idma64 : iDMA64 core driver
snd_seq_device : ALSA sequencer device management
snd_timer : ALSA timer interface

virt_dma
cfg80211 : wireless configuration support
snd : Advanced Linux Sound Architecture driver for soundcards.
acer_wireless : Acer Wireless Radio Control Driver
acpi_pad : ACPI Processor Aggregator Driver
soundcore : Core sound module
mac_hid
intel_pch_thermal : Intel PCH Thermal driver
sch_fq_codel
parport_pc : PC-style parallel port driver
ppdev
lp
parport
ip_tables : IPv4 packet filter
x_tables : {ip,ip6,arp,eb}_tables backend module
autofs4
rtsx_pci_sdmmc : Realtek PCI-E SD/MMC Card Host Driver
r8169 : RealTek RTL-8169 Gigabit Ethernet driver
ahci : AHCI SATA low-level driver
rtsx_pci : Realtek PCI-E Card Reader Driver
realtek : Realtek PHY driver
libahci : Common AHCI SATA low-level routines
i2c_hid : HID over I2C core driver
hid
wmi : ACPI-WMI Mapping Driver
video : ACPI Video Driver
pinctrl_cannonlake : Intel Cannon Lake PCH pinctrl/GPIO driver
pinctrl_intel : Intel pinctrl/GPIO core driver

File systems :

-Mounted File Systems-

udev	/dev	0.00 % (3.8 GiB of 3.8 GiB)
tmpfs	/run	0.45 % (776.7 MiB of 780.2 MiB)
/dev/sdb4	/	53.23 % (21.8 GiB of 46.7 GiB)
tmpfs	/dev/shm	2.93 % (3.7 GiB of 3.8 GiB)
tmpfs	/run/lock	0.08 % (5.0 MiB of 5.0 MiB)
tmpfs	/sys/fs/cgroup	0.00 % (3.8 GiB of 3.8 GiB)
/dev/loop0	/snap/gnome-3-28-1804/128	100.00 % (0.0 B of 161.5 MiB)
/dev/loop3	/snap/arduino-mhall119/5	100.00 % (0.0 B of 151.4 MiB)
/dev/loop2	/snap/core20/975	100.00 % (0.0 B of 61.8 MiB)
/dev/loop1	/snap/core/10958	100.00 % (0.0 B of 99.2 MiB)
/dev/loop6	/snap/wine-platform-3-stable/10	100.00 % (0.0 B of 74.0 MiB)
/dev/loop5	/snap/gnome-characters/708	100.00 % (0.0 B of 384.0 KiB)
/dev/loop8	/snap/gnome-logs/100	100.00 % (0.0 B of 1.0 MiB)
/dev/loop7	/snap/arduino-mhall119/7	100.00 % (0.0 B of 178.2 MiB)
/dev/loop10	/snap/gnome-3-34-1804/66	100.00 % (0.0 B of 219.0 MiB)
/dev/loop9	/snap/core/10908	100.00 % (0.0 B of 99.2 MiB)
/dev/loop11	/snap/sublime-text/85	100.00 % (0.0 B of 67.6 MiB)
/dev/loop4	/snap/postman/133	100.00 % (0.0 B of 175.4 MiB)
/dev/loop16	/snap/core20/904	100.00 % (0.0 B of 61.8 MiB)
/dev/loop17	/snap/wine-platform-3-stable/11	100.00 % (0.0 B of 99.5 MiB)
/dev/loop14	/snap/gtk2-common-themes/13	100.00 % (0.0 B of 256.0 KiB)
/dev/loop12	/snap/core18/1997	100.00 % (0.0 B of 55.5 MiB)
/dev/loop13	/snap/wine-platform-runtime/216	100.00 % (0.0 B of 337.8 MiB)
/dev/loop15	/snap/core18/1988	100.00 % (0.0 B of 55.5 MiB)
/dev/loop19	/snap/gnome-logs/103	100.00 % (0.0 B of 640.0 KiB)
/dev/loop22	/snap/gnome-characters/570	100.00 % (0.0 B of 384.0 KiB)
/dev/loop23	/snap/gnome-3-28-1804/145	100.00 % (0.0 B of 162.9 MiB)
/dev/loop26	/snap/gnome-calculator/826	100.00 % (0.0 B of 2.5 MiB)
/dev/loop21	/snap/code/62	100.00 % (0.0 B of 199.5 MiB)
/dev/loop18	/snap/gnome-3-26-1604/102	100.00 % (0.0 B of 140.8 MiB)
/dev/loop28	/snap/gnome-system-monitor/148	100.00 % (0.0 B of 2.2 MiB)
/dev/loop24	/snap/gtk-common-themes/1515	100.00 % (0.0 B of 65.1 MiB)
/dev/loop30	/snap/gnome-3-26-1604/100	100.00 % (0.0 B of 140.8 MiB)
/dev/loop25	/snap/postman/132	100.00 % (0.0 B of 175.4 MiB)
/dev/loop29	/snap/discord/121	100.00 % (0.0 B of 75.8 MiB)
/dev/loop31	/snap/gtk-common-themes/1514	100.00 % (0.0 B of 64.9 MiB)
/dev/loop27	/snap/wine-platform-runtime/212	100.00 % (0.0 B of 337.5 MiB)

/dev/loop20	/snap/spotify/46	100.00 % (0.0 B of 179.6 MiB)
/dev/loop37	/snap/gtk2-common-themes/9	100.00 % (0.0 B of 256.0 KiB)
/dev/loop35	/snap/discord/122	100.00 % (0.0 B of 76.8 MiB)
/dev/loop38	/snap/sublime-text/97	100.00 % (0.0 B of 57.2 MiB)
/dev/loop33	/snap/gnome-system-monitor/157	100.00 % (0.0 B of 2.2 MiB)
/dev/loop36	/snap/gnome-3-34-1804/60	100.00 % (0.0 B of 218.0 MiB)
/dev/loop34	/snap/spotify/45	100.00 % (0.0 B of 179.5 MiB)
/dev/loop39	/snap/gnome-calculator/884	100.00 % (0.0 B of 2.5 MiB)
/dev/loop32	/snap/code/61	100.00 % (0.0 B of 199.5 MiB)
/dev/sda1	/boot/efi	57.11 % (41.2 MiB of 96.0 MiB)
/dev/sdb5	/home	97.02 % (2.9 GiB of 97.1 GiB)
tmpfs	/run/user/121	0.00 % (780.2 MiB of 780.2 MiB)
tmpfs	/run/user/1001	0.00 % (780.2 MiB of 780.2 MiB)

Processor :

-Processors-

Package Information

Intel(R) Core(TM) i5-8300H CPU @ 2.30GHz	0	0:0	4000.00 MHz
Intel(R) Core(TM) i5-8300H CPU @ 2.30GHz	1	0:1	4000.00 MHz
Intel(R) Core(TM) i5-8300H CPU @ 2.30GHz	2	0:2	4000.00 MHz
Intel(R) Core(TM) i5-8300H CPU @ 2.30GHz	3	0:3	4000.00 MHz
Intel(R) Core(TM) i5-8300H CPU @ 2.30GHz	4	0:0	4000.00 MHz
Intel(R) Core(TM) i5-8300H CPU @ 2.30GHz	5	0:1	4000.00 MHz
Intel(R) Core(TM) i5-8300H CPU @ 2.30GHz	6	0:2	4000.00 MHz
Intel(R) Core(TM) i5-8300H CPU @ 2.30GHz	7	0:3	4000.00 MHz

Memory :

-Memory-

MemTotal	Total Memory	7989672 KiB
MemFree	Free Memory	2743960 KiB
MemAvailable		5172596 KiB
Buffers		661508 KiB
Cached		1760684 KiB
SwapCached	Cached Swap	0 KiB
Active		3336996 KiB
Inactive		1047832 KiB
Active(anon)		1973904 KiB
Inactive(anon)		134288 KiB
Active(file)		1363092 KiB
Inactive(file)		913544 KiB
Unevictable		7580 KiB
Mlocked		48 KiB
SwapTotal	Virtual Memory	8000508 KiB
SwapFree	Free Virtual Memory	8000508 KiB
Dirty		464 KiB
Writeback		0 KiB
AnonPages		1970444 KiB
Mapped		635620 KiB
Shmem		145564 KiB
KReclaimable		460776 KiB
Slab		579676 KiB
SReclaimable		460776 KiB
SUnreclaim		118900 KiB
KernelStack		14880 KiB
PageTables		65216 KiB
NFS_Unstable		0 KiB
Bounce		0 KiB
WritebackTmp		0 KiB
CommitLimit		11995344 KiB
Committed_AS		9248904 KiB
VmallocTotal		-1 KiB
VmallocUsed		63088 KiB
VmallocChunk		0 KiB
Percpu		3200 KiB
HardwareCorrupted		0 KiB
AnonHugePages		0 KiB
ShmemHugePages		0 KiB
ShmemPmdMapped		0 KiB

FileHugePages		0 KiB
FilePmdMapped		0 KiB
CmaTotal	0 KiB	
CmaFree	0 KiB	
HugePages_Total		0
HugePages_Free		0
HugePages_Rsvd		0
HugePages_Surp		0
Hugepagesize		2048 KiB
Hugetlb	0 KiB	
DirectMap4k	672444 KiB	
DirectMap2M		7555072 KiB
DirectMap1G		0 KiB

PCI devices :

-PCI Devices-

Host bridge	: Intel Corporation Device 3e10 (rev 07)
PCI bridge	: Intel Corporation Xeon E3-1200 v5/E3-1500 v5/6th Gen Core Processor PCIe Controller (x16) (rev 07) (prog-if 00 [Normal decode])
VGA compatible controller	: Intel Corporation Device 3e9b (prog-if 00 [VGA controller])
System peripheral	: Intel Corporation Xeon E3-1200 v5/v6 / E3-1500 v5 / 6th/7th Gen Core Processor Gaussian Mixture Model
Signal processing controller	: Intel Corporation Cannon Lake PCH Thermal Controller (rev 10)
USB controller	: Intel Corporation Cannon Lake PCH USB 3.1 xHCI Host Controller (rev 10) (prog-if 30 [XHCI])
RAM memory	: Intel Corporation Cannon Lake PCH Shared SRAM (rev 10)
Network controller	: Intel Corporation Wireless-AC 9560 [Jefferson Peak] (rev 10)

USB devices :

-USB Devices-
Linux Foundation 3.0 root hub
Quanta Computer, Inc.
Intel Corp.
Linux Foundation 2.0 root hub

Battery :

-Battery: BAT1-

State	: Full
Capacity	: 100 / Full
Battery Technology	: Li-ion
Manufacturer	: LG
Model Number	: PABAS0241231
Serial Number	: 41167

Sensors :

-Sensors-

../../BAT1/in0	Voltage	16.31V	
../../thermal_zone0/temp1		Temperature	76.00°C
../../thermal_zone1/temp1		Temperature	59.00°C
coretemp/temp1	Temperature	89.00°C	
coretemp/temp2	Temperature	89.00°C	
coretemp/temp3	Temperature	89.00°C	
coretemp/temp4	Temperature	80.00°C	
coretemp/temp5	Temperature	80.00°C	
thermal/thermal_zone2	Temperature	91.00°C	
thermal/thermal_zone0	Temperature	77.00°C	
thermal/thermal_zone1	Temperature	59.00°C	

Storage :

-SCSI Disks-
ATA KINGSTON RBUSNS8
ATA HGST HTS721010A9

DMI :

-Product-

Name	: Predator PH315-51
Family	: Predator Helios 300
Vendor	: Acer (Acer, www.acer.com)
Version	: V1.24

-BIOS-

Date : 12/05/2018
Vendor : Insyde Corp.
Version : V1.24
-Board-
Name : Sienta_CFS
Vendor : CFL
Version : V1.24
Serial Number : (Not available; Perhaps try running HardInfo as root.)
Asset Tag : Type2 - Board Serial Number
-Chassis-
Vendor : Acer (Acer, www.acer.com)
Type : [10] Notebook
Version : V1.24
Serial Number : (Not available; Perhaps try running HardInfo as root.)
Asset Tag : (Not available; Perhaps try running HardInfo as root.)

BenchMarking (GPU Drawing) :

-GPU Drawing-

Intel(R) Core(TM) i5-8300H CPU @ 2.30GHz	8x 4000.00 MHz	13017.75

BenchMarking (CPU Fibonacci) :

-CPU Fibonacci-

Intel(R) Core(TM) i5-8300H CPU @ 2.30GHz	8x 4000.00 MHz	0.63
AMD Phenom(tm) 9500 Quad-Core Processor	4x 1100.00 MHz	3.06
Intel(R) Core(TM)2 Duo CPU T7250@ 2.00GHz	2x 2001.00 MHz	4.57
Intel(R) Pentium(R) 4 CPU 2.80GHz	1x 2791.00 MHz	5.65
AMD Athlon(tm) Processor	1x 1210.00 MHz	7.34
Intel(R) Core(TM)2 CPU T5200@ 1.60GHz	2x 1067.00 MHz	7.27
Intel(R) Atom(TM) CPU330 @ 1.60GHz	4x 1596.00 MHz	8.79
AMD Phenom(tm) 9150e Quad-Core Processor	4x 900.00 MHz	3.28
AMD Turion(tm) 64 Mobile Technology MK-36	1x 800.00 MHz	6.20
AMD Athlon(tm) 64 Processor 3200+	1x 1000.00 MHz	5.76
Intel(R) Core(TM)2 Quad CPUQ9400@ 2.66GHz	4x 2003.00 MHz	2.76
Intel(R) Core(TM)2 Duo CPU T7500@ 2.20GHz	1x 2200.00 MHz	4.09
AMD Athlon(tm) XP	1x 906.00 MHz	33.45
Intel(R) Core(TM)2 Quad CPUQ6700@ 2.66GHz	4x 2669.00 MHz	3.54
Intel(R) Pentium(R) 4 CPU 2.00GHz	1x 1994.00 MHz	8.70
Intel(R) Core(TM)2 Duo CPU E6550@ 2.33GHz	2x 2700.00 MHz	3.66
Intel(R) Core(TM)2 Duo CPU T7100@ 1.80GHz	2x 1801.00 MHz	4.79
Dual Core AMD Opteron(tm) Processor 165	2x 1979.00 MHz	4.26
Intel(R) Pentium(R) 4 CPU 2.40GHz	1x 2400.00 MHz	6.35
Intel(R) Core(TM)2 Duo CPU P8600@ 2.40GHz	1x 2400.00 MHz	3.37
AMD Turion(tm) X2 Dual-Core Mobile RM-70	2x 500.00 MHz	4.97
Intel(R) Pentium(R) 4 CPU 2.53GHz	1x 2525.00 MHz	10.36
Unknown CPU Type	1x 1660.00 MHz	5.73
Pentium III (Coppermine)	2x 999.00 MHz	10.22
Intel(R) Core(TM)2 CPU T5600@ 1.83GHz	2x 1000.00 MHz	5.24
Intel(R) Celeron(R) CPU 3.06GHz	1x 3059.00 MHz	3.96
AMD Phenom(tm) 9850 Quad-Core Processor	4x 1300.00 MHz	2.59
Intel(R) Pentium(R) 4 CPU 3.00GHz	2x 3458.00 MHz	4.98
AMD Sempron(tm) Processor LE-1200	1x 2109.00 MHz	6.08
Intel(R) Pentium(R) 4 CPU 2.80GHz	1x 2791.00 MHz	5.65
AMD Athlon(tm) 64 X2 Dual Core Processor 4000+	2x 1000.00 MHz	4.12
AMD Athlon(tm) XP 1600+	1x 1398.00 MHz	6.39
Genuine Intel(R) CPU T2050@ 1.60GHz	2x 800.00 MHz	6.10
Intel(R) Celeron(R) CPU560@ 2.13GHz	1x 2128.00 MHz	4.78
AMD Phenom(tm) 8650 Triple-Core Processor	3x 2300.00 MHz	3.09
Intel(R) Core(TM) i7 CPU 920@ 2.67GHz	8x 3799.00 MHz	2.79
AMD Athlon(tm) 64 X2 Dual Core Processor 5000+	2x 1000.00 MHz	3.11
AMD Athlon(tm) XP 2500+	1x 1792.00 MHz	7.18
Intel(R) Xeon(R) CPU3040@ 1.86GHz	2x 1862.00 MHz	4.81
AMD Athlon(tm) X2 Dual-Core QL-60	2x 1900.00 MHz	4.35
Intel(R) Core(TM)2 CPU T7400@ 2.16GHz	2x 2161.00 MHz	4.98
AMD Turion(tm) 64 X2 Mobile Technology TL-60	2x 2000.00 MHz	4.19
AMD Turion(tm) 64 X2 Mobile Technology TL-52	2x 1600.00 MHz	5.45
AMD Athlon(tm) 64 X2 Dual Core Processor 6000+	2x 3051.00 MHz	2.75
Intel(R) Atom(TM) CPU N280 @ 1.66GHz	2x 1000.00 MHz	8.97
Intel(R) Core(TM)2 Quad CPUQ9550@ 2.83GHz	4x 2830.00 MHz	2.72

Intel(R) Core(TM)2 Duo CPU T7300@ 2.00GHz	2x 2001.00 MHz	4.77	
AMD Sempron(tm) Processor 3600+	1x 1000.00 MHz	5.70	
AMD Athlon(tm)	1x 2305.00 MHz	4.47	
AMD Athlon(tm) X2 Dual Core Processor BE-2300	2x 1899.00 MHz	4.37	
Intel(R) Core(TM)2 Duo CPU E6750@ 2.66GHz	2x 2671.00 MHz	4.10	
PowerPC 740/750	1x 280.00 MHz	58.08	

Task-1 [Matrix Multiplication] :

The three implementations of matrix multiplication that I studied are:

- IJK
- JKI
- KIJ

The difference between the above implementations is the order of looping through the given matrices. The calculations below were done on square matrices of size 500. The data for the matrices were generated through a random number generator.

```
void ijk(int m){
  int sum = 0;
  for (int i = 0; i<m; i++){
    for(int j = 0; j<m; j++){
      for(int k = 0; k<m; k++){
        sum += A[i][k]*B[k][j];
      }
      M[i][j] = sum;
      sum = 0;
    }
  }
}
```

```
void jki(int m){
  for (int j = 0; j<m; j++){
    for(int k = 0; k<m; k++){
      int r = B[k][j];
      for(int i = 0; i<m; i++){
        M[i][j] += A[i][k]*r;
      }
    }
  }
}
```

```
void kij(int m){
  for (int k = 0; k<m; k++){
    for(int i = 0; i<m; i++){
      int r = A[i][k];
      for(int j = 0; j<m; j++){
        M[i][j] += B[k][j]*r;
      }
    }
  }
}
```

i) IJK

Here, a row of Matrix A and a column of B is loaded into the memory. M[i][j] is loaded into the fast and slow memories.

```
==14739== Cachegrind, a cache and branch-prediction profiler
==14739== Copyright (C) 2002-2017, and GNU GPL'd, by Nicholas Nethercote et al.
==14739== Using Valgrind-3.13.0 and LibVEX; rerun with -h for copyright info
==14739== Command: ./a.out
==14739==
--14739-- warning: L3 cache found, using its data for the LL simulation.
==14739==
==14739== I   refs:      3,045,180,797
==14739== I1  misses:            1,027
==14739== LLi misses:            1,009
==14739== I1  miss rate:          0.00%
==14739== LLi miss rate:          0.00%
==14739==
==14739== D   refs:      1,269,319,733  (1,264,056,442 rd   + 5,263,291 wr)
==14739== D1  misses:       62,968,656  (   62,695,395 rd   +   273,261 wr)
==14739== LLd misses:           50,630  (        2,067 rd   +    48,563 wr)
==14739== D1  miss rate:           5.0% (          5.0%     +       5.2%  )
==14739== LLd miss rate:           0.0% (          0.0%     +       0.9%  )
==14739==
==14739== LL refs:          62,969,683  (   62,696,422 rd   +   273,261 wr)
==14739== LL misses:            51,639  (        3,076 rd   +    48,563 wr)
==14739== LL miss rate:            0.0% (          0.0%     +       0.9%  )
```

ii) JKI

Here, a row of A and a row of B are loaded into the memory. B[k][j] is then loaded into the fast memory.

```
==15024== Cachegrind, a cache and branch-prediction profiler
==15024== Copyright (C) 2002-2017, and GNU GPL'd, by Nicholas Nethercote et al.
==15024== Using Valgrind-3.13.0 and LibVEX; rerun with -h for copyright info
==15024== Command: ./a.out
==15024==
--15024-- warning: L3 cache found, using its data for the LL simulation.
==15024==
==15024== I   refs:      4,169,931,685
==15024== I1  misses:            1,028
==15024== LLi misses:            1,011
==15024== I1  miss rate:          0.00%
==15024== LLi miss rate:          0.00%
==15024==
==15024== D   refs:      1,644,070,070  (1,514,056,677 rd   + 130,013,393 wr)
==15024== D1  misses:      250,285,305  (  250,252,660 rd   +      32,645 wr)
==15024== LLd misses:           50,630  (       18,067 rd   +      32,563 wr)
==15024== D1  miss rate:          15.2% (         16.5%     +         0.0%  )
==15024== LLd miss rate:           0.0% (          0.0%     +         0.0%  )
==15024==
==15024== LL refs:         250,286,333  (  250,253,688 rd   +      32,645 wr)
==15024== LL misses:            51,641  (       19,078 rd   +      32,563 wr)
==15024== LL miss rate:            0.0% (          0.0%     +         0.0%  )
```

iii) KIJ

Here, a row of A and a row of M are loaded into the memory leading to better temporal locality.

```
==14837== Cachegrind, a cache and branch-prediction profiler
==14837== Copyright (C) 2002-2017, and GNU GPL'd, by Nicholas Nethercote et al.
==14837== Using Valgrind-3.13.0 and LibVEX; rerun with -h for copyright info
==14837== Command: ./a.out
==14837==
--14837-- warning: L3 cache found, using its data for the LL simulation.
==14837==
==14837== I   refs:      4,169,931,685
==14837== I1  misses:            1,028
==14837== LLi misses:            1,011
==14837== I1  miss rate:          0.00%
==14837== LLi miss rate:          0.00%
==14837==
==14837== D   refs:      1,644,070,070  (1,514,056,677 rd   + 130,013,393 wr)
==14837== D1  misses:        8,301,305  (    8,268,660 rd   +      32,645 wr)
==14837== LLd misses:           50,630  (       18,067 rd   +      32,563 wr)
==14837== D1  miss rate:           0.5% (          0.5%     +         0.0%  )
==14837== LLd miss rate:           0.0% (          0.0%     +         0.0%  )
==14837==
==14837== LL refs:           8,302,333  (    8,269,688 rd   +      32,645 wr)
==14837== LL misses:            51,641  (       19,078 rd   +      32,563 wr)
==14837== LL miss rate:            0.0% (          0.0%     +         0.0%  )
```

Since the LLd and LL miss rates are 0 in all cases, let's compare the D1 miss rate for the 3 cases. Upon comparing the D1 miss rate of the 3 implementations we find that KIJ is the best implementation while JKI is the worst.

D1 miss rates :

IJK - 5.0%

JKI - 15.2%

KIJ - 0.5%

For an even better comparison, let's run all three implementations together and look at the relative memory usage. The below output was generated by running the implementations with square matrixes of size 2000.

```
Samples: 2M of event 'cycles', Event count (approx.): 599488522640
  Children      Self  Command  Shared Object      Symbol
+  100.00%     0.00%  a.out    [unknown]          [.] 0x10a6258d4c544155
+  100.00%     0.00%  a.out    libc-2.27.so       [.] __libc_start_main
+   99.97%     0.03%  a.out    a.out              [.] main
+   54.20%    54.03%  a.out    a.out              [.] jki
+   30.84%    30.73%  a.out    a.out              [.] ijk
+   14.90%    14.86%  a.out    a.out              [.] kij
```

Hence, KIJ is the best implementation.

Task-2 [Merge Sort] :

Let us begin by finding the Miss rates of the default implementation. All the calculations were done with an array of size $5*10^6$, generated using a Random Number Generator.

```
==6430== Cachegrind, a cache and branch-prediction profiler
==6430== Copyright (C) 2002-2017, and GNU GPL'd, by Nicholas Nethercote et al.
==6430== Using Valgrind-3.13.0 and LibVEX; rerun with -h for copyright info
==6430== Command: ./a.out
==6430==
--6430-- warning: L3 cache found, using its data for the LL simulation.
==6430==
==6430== I   refs:      6,227,549,310
==6430== I1  misses:            1,146
==6430== LLi misses:            1,134
==6430== I1  miss rate:          0.00%
==6430== LLi miss rate:          0.00%
==6430==
==6430== D   refs:      3,114,838,327  (2,726,603,044 rd   + 388,235,283 wr)
==6430== D1  misses:       13,971,800  (    6,879,100 rd   +   7,092,700 wr)
==6430== LLd misses:        4,008,140  (    1,897,178 rd   +   2,110,962 wr)
==6430== D1  miss rate:           0.4% (          0.3%     +         1.8%  )
==6430== LLd miss rate:           0.1% (          0.1%     +         0.5%  )
==6430==
==6430== LL refs:          13,972,946  (    6,880,246 rd   +   7,092,700 wr)
==6430== LL misses:         4,009,274  (    1,898,312 rd   +   2,110,962 wr)
==6430== LL miss rate:            0.0% (          0.0%     +         0.5%  )
```

Tiled Merged Sort

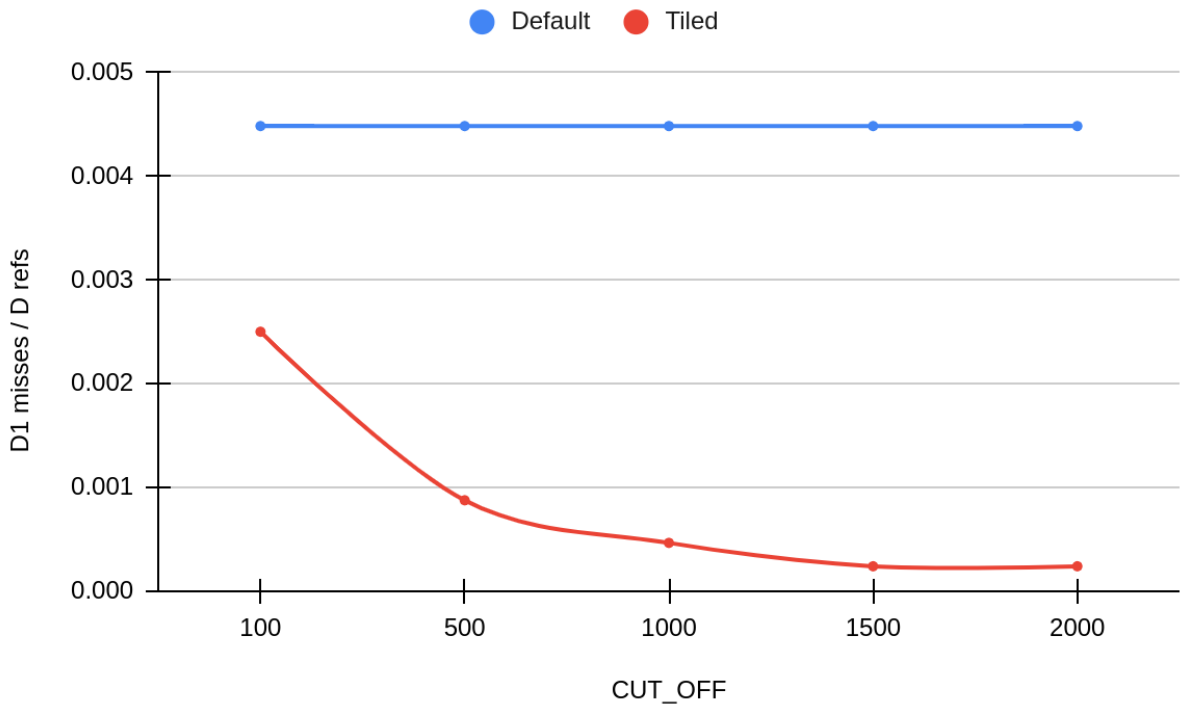
To improve the Cache performance, Bubble Sort was implemented with has much lower cache usage as it only uses two extra pointers to sort the array. However, it sacrifices computational time in exchange. Bubble sort has time complexity $O[n^2]$ while Merge Sort has $O[n*\log_2(n)]$. Here, the mergesort function is changed to run BubbleSort once the array is smaller than a CUT_OFF.

```
void mergeSort(int arr[], int l, int r){
    if (r-l < CUT_OFF){
        bubble_sort(arr,l,r);
        return;
    }
    if (l < r) {
        // Same as (l+r)/2, but avoids overflow for
        // large l and h
        int m = l + (r - l) / 2;

        // Sort first and second halves
        mergeSort(arr, l, m);
        mergeSort(arr, m + 1, r);

        merge(arr, l, m, r);
    }
}
```

Let us now study how D1 misses change with CUT_OFF.



As you can see above the miss ratio generally decreases with an increase in CUT_OFF. However, there's also a significant increase in computational time. For further analysis, CUT_OFF = 1000.

```
==7190== Cachegrind, a cache and branch-prediction profiler
==7190== Copyright (C) 2002-2017, and GNU GPL'd, by Nicholas Nethercote et al.
==7190== Using Valgrind-3.13.0 and LibVEX; rerun with -h for copyright info
==7190== Command: ./a.out
==7190==
--7190-- warning: L3 cache found, using its data for the LL simulation.
==7190==
==7190== I   refs:      59,154,878,490
==7190== I1  misses:             1,153
==7190== LLi misses:             1,141
==7190== I1  miss rate:           0.00%
==7190== LLi miss rate:           0.00%
==7190==
==7190== D   refs:      29,905,536,858  (26,674,503,233 rd   + 3,231,033,625 wr)
==7190== D1  misses:        13,956,705  (     6,871,133 rd   +     7,085,572 wr)
==7190== LLd misses:         4,008,042  (     1,897,090 rd   +     2,110,952 wr)
==7190== D1  miss rate:            0.0% (           0.0%     +           0.2%  )
==7190== LLd miss rate:            0.0% (           0.0%     +           0.1%  )
==7190==
==7190== LL refs:           13,957,858  (     6,872,286 rd   +     7,085,572 wr)
==7190== LL misses:          4,009,183  (     1,898,231 rd   +     2,110,952 wr)
==7190== LL miss rate:             0.0% (           0.0%     +           0.1%  )
```

Here’s, how memory usage changed upon implementing tiled merge sort.

```
Samples: 11K of event 'cycles', Event count (approx.): 3871646344
  Children      Self  Command  Shared Object      Symbol
+   99.83%     0.00%  a.out    [unknown]          [k] 0x113e258d4c544155
+   99.83%     0.00%  a.out    libc-2.27.so       [.] __libc_start_main
+   97.52%     0.99%  a.out    a.out              [.] main
+   96.04%     3.92%  a.out    a.out              [.] mergeSort
+   92.22%    91.59%  a.out    a.out              [.] merge
```

```
Samples: 68K of event 'cycles', Event count (approx.): 22608389455
  Children      Self  Command  Shared Object      Symbol
+   99.97%     0.00%  a.out    [unknown]          [.] 0x100e258d4c544155
+   99.97%     0.00%  a.out    libc-2.27.so       [.] __libc_start_main
+   99.59%     0.18%  a.out    a.out              [.] main
+   99.32%     0.01%  a.out    a.out              [.] mergeSort
+   90.06%    89.98%  a.out    a.out              [.] bubble_sort
+    9.25%     9.17%  a.out    a.out              [.] merge
```