

Лекция 3

Инструменты отладки и профилирования:

- cuda-gdb
- Data Display Debugger (ddd)
- Nsight Eclipse Plugins
- Nsight Visual Studio Code Edition
- nvprof
- nvvp
- Nsight Compute CLI
- Nsight Compute

```
> ~Lecture3/Lab3-gdb> g++ lab3a.cpp -g3 -o lab3a  
> ~Lecture3/Lab3-gdb> gdb lab3a
```

```
(gdb) list main
```

```
.....  
36         gettimeofday(&t, NULL);  
37         Start =(double)t.tv_sec*1000000.0 +  
         (double)t.tv_usec;  
38         hTest(N,a,b);  
39         gettimeofday(&t, NULL);  
40         Finish =(double)t.tv_sec*1000000.0 +  
         (double)t.tv_usec;
```

```
.....  
(gdb) b 38
```

```
Breakpoint 1 at 0x400865: file lab3a.cpp, line 38.
```

(gdb) **run 256**

Starting program: ~/Lecture3/Lab3-gdb/lab3a 16

(gdb) **step**

hTest (N=16, a=0x613e70, b=0x613ec0) at lab3a.cpp:7

```
7         for(int i=0; i<N;i++)
```

(gdb) list hTest

```
1         #include <malloc.h>
```

```
2         #include <stdio.h>
```

```
3         #include <stdlib.h>
```

```
4         #include <sys/time.h>
```

```
5
```

```
6         void hTest(int N, int* a, int* b){
```

```
7             for(int i=0; i<N;i++)
```

```
8                 a[i]+=b[i];
```

```
9             }
```

```
(gdb) info args
```

```
N = 16
```

```
a = 0x613e70
```

```
b = 0x613ec0
```

```
(gdb) info locals
```

```
i = 0
```

```
(gdb) next 8
```

```
7         for(int i=0; i<N;i++)
```

```
(gdb) info locals
```

```
i = 3
```

```
(gdb) print b[2]
```

```
$1 = 5
```

```
(gdb) print a[2]
```

```
$2 = 9
```

```
gdb) break 8 if i==12
```

```
Breakpoint 3 at 0x400705: file lab3a.cpp, line 8.
```

```
(gdb) c
```

```
Continuing.
```

```
Breakpoint 3, hTest (N=16, a=0x613e70, b=0x613ec0) at  
lab3a.cpp:8
```

```
8          a[i]+=b[i];
```

```
(gdb) info locals
```

```
i = 12
```

```
(gdb) finish
```

```
Run till exit from #0 hTest (N=16, a=0x613e70,  
b=0x613ec0) at lab3a.cpp:8
```

```
main (argc=2, argv=0x7fffffffdd9b8) at lab3a.cpp:39
```

```
39      gettimeofday(&t, NULL);
```

(gdb) **x/16d b**

0x613ec0:	1	3	5	7
0x613ed0:	9	11	13	15
0x613ee0:	17	19	21	23
0x613ef0:	25	27	29	31

(gdb) **x/16d a**

0x613e70:	1	5	9	13
0x613e80:	17	21	25	29
0x613e90:	33	37	41	45
0x613ea0:	49	53	57	61

(gdb) **print a[2]-b[2]**

\$16 = 4

(gdb) **c**

Continuing.

Elapsed time: 9.57138e+06 ms

0	1	1
1	5	3
2	9	5
3	13	7
4	17	9
5	21	11

..... • •

13	53	27
14	57	29
15	61	31

[Inferior 1 (process 4272) exited normally]

(gdb) **quit**

Отладка многопоточных программ

```
~/Lecture3/Lab3-gdb> gdb lab3b
```

```
(gdb) list hTest
```

```
16 void* hTest(void* arg) {  
17     struct targ* s arg=(struct targ*)arg;  
18     int length=s arg->length;  
19     int offset=s_arg->num_thread*length;  
20     int i;  
21     for(i=0;i<length;i++)  
22         a[i+offset]+=/*1000*sin((double)*/b[i+offset];  
23     return NULL;  
25 }
```

```
(gdb) break lab3b.cpp:22
```

```
Breakpoint 1 at 0x40083f: file lab3b.cpp, line 22.
```



```
(gdb) run 4 16
```

```
Starting program: ../Lecture3/Lab3-gdb/lab3b 4 16
```

```
[Thread debugging using libthread_db enabled]
```

```
Using host libthread_db library
```

```
"/lib64/libthread_db.so.1".
```

```
[New Thread 0x7ffff6ed1700 (LWP 10741)]
```

```
[New Thread 0x7ffff66d0700 (LWP 10742)]
```

```
[New Thread 0x7ffff5ecf700 (LWP 10743)]
```

```
[Switching to Thread 0x7ffff6ed1700 (LWP 10741)]
```

```
Thread 2 "lab3b" hit Breakpoint 1, hTest (arg=0x614e70)  
at lab3b.cpp:22
```

```
22          a[i+offset]+=/*1000*sin((double)*/b[i+offset];
```

(gdb) **info threads**

	Id	Target Id	Frame
1	Thread 0x7ffff7fc0740	(LWP 10737) "lab3b"	clone () at ../sysdeps/unix/sysv/linux/x86_64/clone.S:78
* 2	Thread 0x7ffff6ed1700	(LWP 10741) "lab3b"	hTest (arg=0x614e70) at lab3b.cpp:22
3	Thread 0x7ffff66d0700	(LWP 10742) "lab3b"	hTest (arg=0x614e7c) at lab3b.cpp:22
4	Thread 0x7ffff5ecf700	(LWP 10743) "lab3b"	clone () at ../sysdeps/unix/sysv/linux/x86_64/clone.S:78

```
(gdb) print offset
```

```
$1 = 0
```

```
(gdb) thread 3
```

```
[Switching to thread 3 (Thread 0x7ffff66d0700 (LWP  
10742))]
```

```
#0  hTest (arg=0x614e7c) at lab3b.cpp:22
```

```
22
```

```
a[i+offset] += /*1000*sin((double)*/b[i+offset];
```

```
(gdb) print offset
```

```
$2 = 4
```

```
(gdb) break 22 thread 3
```

Note: breakpoint 1 (all threads) also set at pc
0x40083f.

Breakpoint 2 at 0x40083f: file lab3b.cpp, line 22.

```
(gdb) info breakpoints
```

Num	Type	Disp	Enb	Address	What
1	breakpoint	keep y		0x0000000000040083f	in
hTest(void*) at lab3b.cpp:22					
breakpoint already hit 1 time					
2	breakpoint	keep y		0x0000000000040083f	in
hTest(void*) at lab3b.cpp:22 thread 3					
stop only in thread 3					

```
(gdb) delete 1
```

```
(gdb) x/16d a
```

0x614ee0:	0	2	4	6
0x614ef0:	8	10	12	14
0x614f00:	16	18	20	22
0x614f10:	24	26	28	30

```
(gdb) continue
```

```
Continuing.
```

```
Thread 3 "lab3b" hit Breakpoint 2, hTest (arg=0x614e7c)
at lab3b.cpp:22
```

```
22          a[i+offset]+=/*1000*sin((double)*/b[i+offset];
```

```
(gdb) c
```

```
.....
(gdb) x/16d a
```

0x614ee0:	1	2	4	6
0x614ef0:	17	21	12	14
0x614f00:	33	37	20	22
0x614f10:	49	53	28	30

(gdb) **c**

Continuing.

[Thread 0x7ffff5ecf700 (LWP 10743) exited]

[Thread 0x7ffff66d0700 (LWP 10742) exited]

Thread-specific breakpoint 2 deleted - thread 3 no longer in the thread list.

[Thread 0x7ffff56ce700 (LWP 11167) exited]

[Thread 0x7ffff6ed1700 (LWP 10741) exited]

Elapsed time: 2.22941e+06 ms

0	1	1
1	3	5
2	5	9

.....

14	29	57
15	31	61

[Inferior 1 (process 10737) exited normally]

```
(gdb) print a[1]
```

```
..... •
```

```
(gdb) info locals
```

```
..... •
```

```
(gdb) info args
```

```
arg = 0x614e7c
```

```
(gdb) print ((struct targ*)arg)->length
```

```
$1 = 4
```


Отладка программ, выполняемых на GPU

<https://docs.nvidia.com/cuda/archive/11.2.0/cuda-gdb/index.html>

```
> nvcc -g -G ...
```

```
~/Workshop/VSC> cuda-gdb lab3c
```

```
(cuda-gdb) list main
```

```
23  __global__ void gSum(int* a, int *b){  
24      int i=threadIdx.x+blockIdx.x*blockDim.x;  
25      a[i]+=b[i];  
26  }
```

```
27
```

```
28  int main(){  
29      int N=VECTOR_LENGTH;  
30      int *a, *b;  
31      int *a_h;  
32
```

```
(cuda-gdb) break 25
```

```
Breakpoint 1 at 0x403fdd: file lab3c.cu, line 26.
```

```
(cuda-gdb) run
```

(cuda-gdb) **info cuda threads**

BlockIdx	ThreadIdx	To	BlockIdx	ThreadIdx	Count	VirtualPC
						Filename Line
Kernel 0						
*	(0,0,0)	(0,0,0)	(0,0,0)	(3,0,0)	4	
0x00007fffe525c3e0						
.../Lecture3/Lab3-cuda-gdb/lab3c.cu				9		
	(1,0,0)	(0,0,0)	(3,0,0)	(3,0,0)	12	
0x00007fffe525c2b0						
.../Lecture3/Lab3-cuda-gdb/lab3c.cu				8		

```
cuda-gdb) cuda block 2 thread 3
```

```
[Switching focus to CUDA kernel 0, grid 1, block  
(2,0,0), thread (3,0,0), device 0, sm 2, warp 0, lane 3]
```

```
8          b[i]=2*i+1;
```

```
(cuda-gdb) print i
```

```
$2 = 11
```

```
(cuda-gdb) n
```

```
9          }
```

```
(cuda-gdb) x/16d b
```

```
0x7fffecc00200: 1          3          5          7
```

```
0x7fffecc00210: 0          0          0          0
```

```
0x7fffecc00220: 0          0          0          0
```

```
0x7fffecc00230: 0          0          0          0
```

```
malkov@192:~> ssh cyber.sibsutis.ru
```

```
malkov@linux-47dw: ~/WORKSHOP/PGP-2023> cuda-gdb lab3c
```

```
(cuda-gdb) break 8
```

```
Breakpoint 1 at 0x403851: file lab3c.cu, line 8.
```

```
(cuda-gdb) run
```

```
Starting program: /home/malkov/WORKSHOP/PGP-2023/lab3c
```

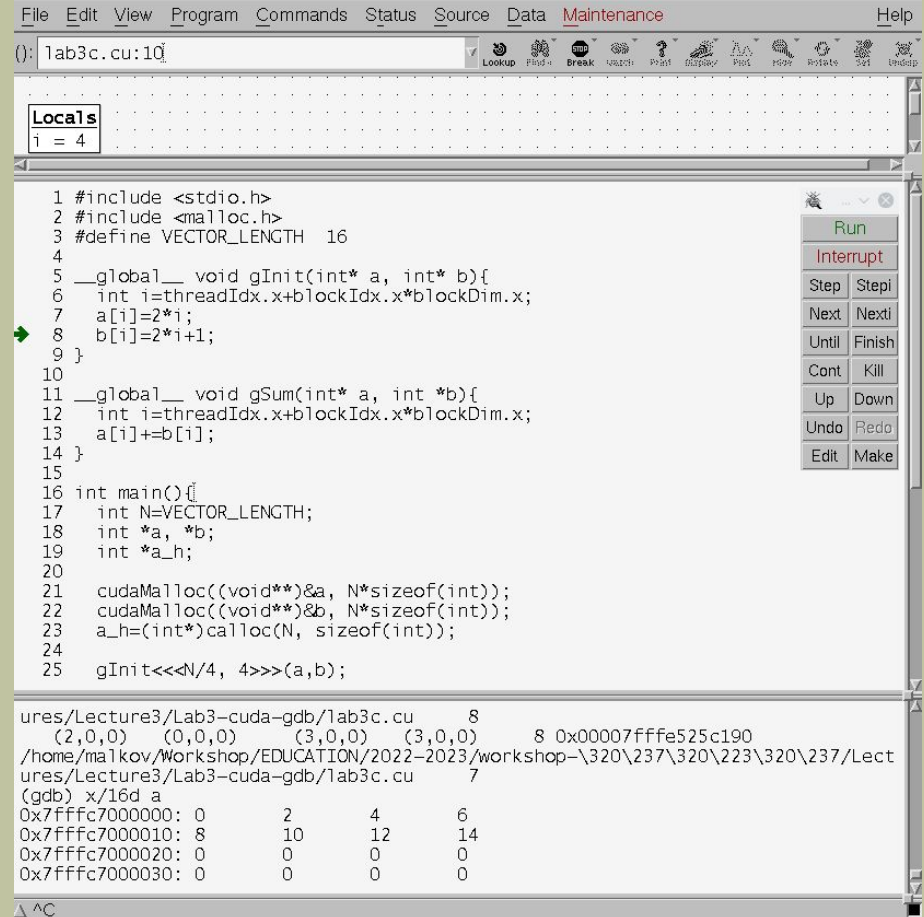
```
.....  
[Switching focus to CUDA kernel 0, grid 1, block  
(0,0,0), thread (0,0,0), device 0, sm 0, warp 0, lane 0]  
Thread 1 "lab3c" hit Breakpoint 1,  
gInit<<<(4,1,1),(4,1,1)>>> (a=0x7fffe6800000,  
b=0x7fffe6800200) at lab3c.cu:8  
8           b[i]=2*i+1;
```

Data Display Debugger (ddd)

```
.../Lecture3/  
Lab3-cuda-gdb>  
ddd cuda-gdb lab3c
```

```
malkov@192:~> ssh  
cyber.sibsutis.ru -X
```

```
malkov@linux-47dw:  
~/WORKSHOP/PGP-2023>  
ddd cuda-gdb lab3c
```



The screenshot shows the Data Display Debugger (ddd) interface. The top menu bar includes File, Edit, View, Program, Commands, Status, Source, Data, Maintenance, and Help. The toolbar contains icons for various debugging actions like LookUp, Find, Break, Watch, Print, Display, Step, Next, Until, Cont, Up, Down, Undo, Redo, Edit, and Make. The main window displays the source code of a CUDA program (lab3c.cu) with line numbers 1 through 25. A green arrow points to line 8, indicating the current execution point. The 'Locals' panel on the left shows the variable 'i' with a value of 4. The bottom panel displays the memory dump for the variable 'a' at address 0x7fffc7000000, showing a 4x4 grid of values.

```
1 #include <stdio.h>  
2 #include <malloc.h>  
3 #define VECTOR_LENGTH 16  
4  
5 __global__ void gInit(int* a, int* b){  
6     int i=threadIdx.x+blockIdx.x*blockDim.x;  
7     a[i]=2*i;  
8     b[i]=2*i+1;  
9 }  
10  
11 __global__ void gSum(int* a, int* b){  
12     int i=threadIdx.x+blockIdx.x*blockDim.x;  
13     a[i]+=b[i];  
14 }  
15  
16 int main(){  
17     int N=VECTOR_LENGTH;  
18     int *a, *b;  
19     int *a_h;  
20  
21     cudaMalloc((void**)&a, N*sizeof(int));  
22     cudaMalloc((void**)&b, N*sizeof(int));  
23     a_h=(int*)calloc(N, sizeof(int));  
24  
25     gInit<<<N/4, 4>>>>(a,b);
```

Memory dump for variable 'a' (x/16d):

Address	Value	Address	Value
0x7fffc7000000: 0	2	0x7fffc7000010: 8	10
0x7fffc7000010: 8	10	0x7fffc7000020: 0	0
0x7fffc7000020: 0	0	0x7fffc7000030: 0	0
0x7fffc7000030: 0	0		

Nsight Eclipse Plugins

The screenshot displays the Nsight Eclipse IDE interface for debugging a CUDA application. The main components are:

- Project Explorer:** Shows the project structure for 'lab3 [C/C++ Application]'. It includes a 'gSum' target on a NVIDIA GeForce RTX 2060. The 'All Kernel Threads' section is expanded, showing various CUDA threads and blocks.
- Source Code Editor:** Displays the file 'vector_types.h'. The code defines a vector length and implements a global sum function. A breakpoint is set at line 13, which is highlighted in yellow.
- Variable Window:** Shows the state of variables at the current execution point. The variable 'b' is expanded to show its details, including its type, value, and memory representation.
- Console Window:** Displays the execution log, including thread creation and the current state of the program when the breakpoint was hit.

Source Code (vector_types.h):

```
#include <stdio.h>
#include <malloc.h>
#define VECTOR_LENGTH 16

__global__ void gInit(int* a, int* b){
    __global__ void gSum(int* a, int* b){
        int i=threadIdx.x+blockIdx.x*blockDim.x;
        a[i]+=b[i];
    }
}

int main(){
    int N=VECTOR_LENGTH;
    int *a, *b;
    int *a_h;

    cudaMalloc((void**)&a, N*sizeof(int));
    cudaMalloc((void**)&b, N*sizeof(int));
    a_h=(int*)calloc(N, sizeof(int));

    gInit<<<N/4, 4>>>(a,b);
    cudaDeviceSynchronize();
    gSum<<<N/4, 4>>>(a,b);
    cudaDeviceSynchronize();
}
```

Variable Window:

Name	Type	T(0.0,0)B(0.0,0)
a	@generic int * @int [16]	0x7ffff7000000
b	@generic int [16]	1 (Decimal)
b[0]	@generic int	0x7ffff7000200
b[1]	@generic int	1
b[2]	@generic int	3
b[3]	@generic int	5
b[4]	@generic int	7
b[5]	@generic int	9
b[6]	@generic int	11
b[7]	@generic int	13
b[8]	@generic int	15
b[9]	@generic int	17
b[10]	@generic int	19
b[11]	@generic int	21
b[12]	@generic int	23

Console Window:

```
lab3 [C/C++ Application] ./usr/local/cuda-11.2/bin/cuda-gdb (8.3.1)
[New Thread 0x7ffff36ef700 (LWP 11136)]
[New Thread 0x7ffff2eee700 (LWP 11137)]
[New Thread 0x7ffff238a700 (LWP 11138)]

Thread 1 "lab3" hit Breakpoint 1, gSum<<<(4,1,1),(4,1,1)>>> (a=0x7ffff7000000, b=0x7ffff7000200) at ../lab3-e.cu:13
13     a[i]+=b[i];
    n
    n
14 }
```

nvprof и Nsight Compute CLI


```
/Lecture3/Lab3-cuda-gdb # ncu --target-processes all ./lab3c
```

```
gInit(int *, int *), 2023-Feb-13 15:09:06, Context 1, Stream 7
```

```
Section: GPU Speed Of Light Throughput
```

DRAM Frequency	cycle/nsecond	6.40
SM Frequency	cycle/nsecond	1.29
Elapsed Cycles	cycle	3,327
Memory [%]	%	1.10
DRAM Throughput	%	0.02
Duration	usecond	2.56

WRN This kernel grid is too small to fill the available resources on this device, resulting in only 0.0 full waves across all SMs. Look at Launch Statistics for more details.

.....

```
/Lecture3/Lab3-cuda-gdb # ncu
```

```
--metrics gpu__time_duration.sum ./lab3c
```

```
gInit(int *, int *), 2023-Feb-13 18:42:52, Context 1, Stream 7
```

```
Section: Command line profiler metrics
```

```
-----
```

```
gpu  time duration.sum    usecond                29.50
```

```
-----
```

```
gSum(int *, int *), 2023-Feb-13 18:42:52, Context 1, Stream 7
```

```
Section: Command line profiler metrics
```

```
-----
```

```
gpu  time duration.sum    usecond                37.57
```

```
-----
```

```
/Lecture3/Lab3-cuda-gdb> nvprof --query-metrics  
===== Warning: Skipping profiling on device 0 since  
profiling is not supported on devices with compute  
capability 7.5 and higher.
```

Use NVIDIA Nsight Compute for GPU profiling and NVIDIA Nsight Systems for GPU tracing and CPU sampling.

Refer <https://developer.nvidia.com/tools-overview> for more details.

```
ip-011@linux-47dw:/home/malkov/WORKSHOP/PGP-2023>
nvprof --query-metrics | less
```

Available Metrics:	Name	Description
Device 0 (GeForce GTX 1050):		
inst_per_warp:	Average number of instructions executed by each warp	
warp_execution_efficiency:	Ratio of the average active threads per warp to the maximum number of threads per warp supported on a multiprocessor	
.....		
gld_transactions_per_request:	Average number of global memory load transactions performed for each global memory load.	
gst_transactions_per_request:	Average number of global memory store transactions performed for each global memory store	
.....		

```
ip-011@linux-47dw:/home/malkov/WORKSHOP/PGP-2023>
```

```
nvprof -m gst_throughput ./lab3c
```

Invocations	Metric Name	Metric Description	Min	Max	Avg
Device "GeForce GTX 1050 (0)"					
Kernel: gSum(int*, int*)					
1	gst_throughput	Global Store Throughput	40.582MB/s	40.582MB/s	40.582MB/s
Kernel: gInit(int*, int*)					
1	gst_throughput	Global Store Throughput	71.303MB/s	71.303MB/s	71.302MB/s

```
/Lecture3/Lab3-cuda-gdb # ncu --list-sections
```

```
/Lecture3/Lab3-cuda-gdb # ncu --query-metrics
```

<https://docs.nvidia.com/nsight-compute/NsightComputeCli/index.html#nvprof-metric-collection>

```
/Lecture3/Lab3-cuda-gdb # ncu --metrics
```

```
lltex  t_bytes_pipe_lsu_mem_global_op_st.sum.per_second  
./lab3c
```

```
gInit(int *, int *), 2023-Feb-13 17:13:20, Context 1, Stream 7
```

```
Section: Command line profiler metrics
```

```
-----  
lltex  t bytes pipe lsu mem global op st.sum.per_second  
Mbyte/second                               89.89
```

```
-----  
gSum(int *, int *), 2023-Feb-13 17:13:20, Context 1, Stream 7
```

```
Section: Command line profiler metrics
```

```
-----  
lltex  t bytes pipe lsu mem global op st.sum.per_second  
Mbyte/second                               41.24  
-----
```



```
ip-011@linux-47dw:/home/malkov/WORKSHOP/PGP-2023>
```

```
nvprof -m gld_throughput ./lab3c
```

Invocations	Metric Name	Metric Description	Min	Max	Avg
Device "GeForce GTX 1050 (0)"					
	Kernel: gInit(int*, int*)				
1	gld_throughput	Global Load Throughput	0.0B/s	0.0B/s	0.0B/s
	Kernel: gSum(int*, int*)				
1	gld_throughput	Global Load Throughput	87.694MB/s	87.694MB/s	87.694MB/s

```
/Lecture3/Lab3-cuda-gdb # ncu --metrics
```

```
lltex  t_bytes_pipe_lsu_mem_global_op_ld.sum.per_second  
./lab3c
```

```
gInit(int *, int *), 2023-Feb-13 15:25:41, Context 1, Stream 7
```

```
Section: Command line profiler metrics
```

```
-----
```

```
lltex  t bytes pipe lsu mem global op ld.sum.per_second  
byte/second                                0
```

```
-----
```

```
gSum(int *, int *), 2023-Feb-13 15:25:41, Context 1, Stream 7
```

```
Section: Command line profiler metrics
```

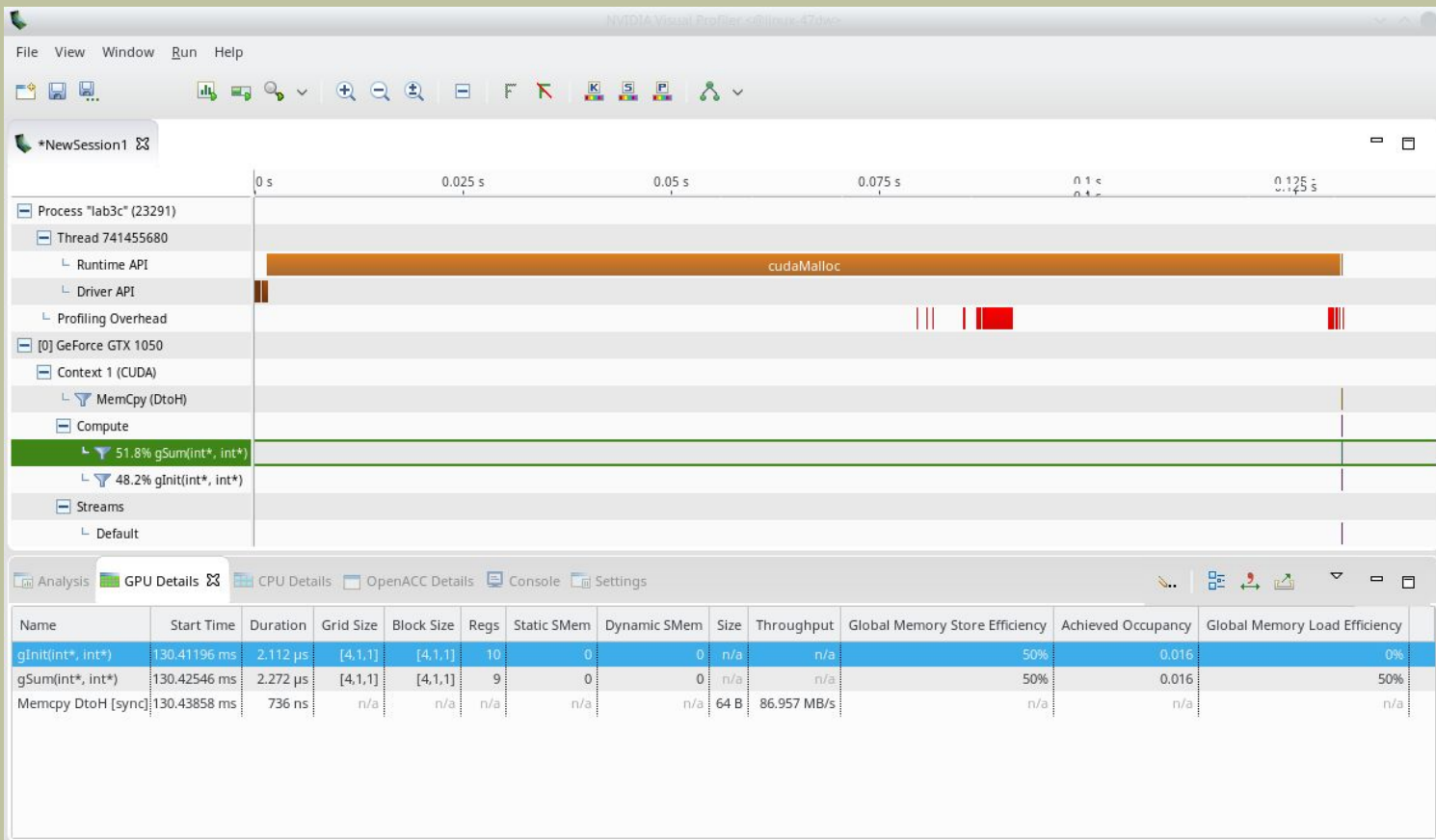
```
-----
```

```
lltex  t bytes pipe lsu mem global op ld.sum.per_second  
Mbyte/second                              82.47
```

```
-----
```

nvvp и Nsight Compute

```
ip-011@linux-47dw: /home/malkov/WORKSHOP/PGP-2023> nvvp ./lab3c
```



0.1 s0.125 s

Achieved Occupan

0.0160%

Metrics and Events <@linux-47dw>

Metrics and Events

Select metrics and events to be collected on individual devices

Device: [0] GeForce GTX 1050

MetricsEvents

☐ Device Memory Write Throughput

☐ Device Memory Write Transactions

☐ ECC Throughput

☐ ECC Transactions

☐ Global Load Throughput

☐ Global Load Transactions

☐ Global Load Transactions Per Request

☒ Global Memory Load Efficiency

☒ Global Memory Store Efficiency

☐ Global Store Throughput

☐ Global Store Transactions

☐ Global Store Transactions Per Request

☐ L2 Read Transactions

☐ L2 Write Transactions

Apply and Run

Cancel

OK

Open a dialog to configure metrics and events, and to run the application to collect

```
/Lecture3/Lab3-cuda-gdb # ncu-ui --target-processes all ./lab3c
```

The screenshot displays the NVIDIA Nsight Compute application window. The top menu bar includes File, Connection, Debug, Profile, Tools, Window, and Help. Below the menu is a toolbar with various icons for connecting, disconnecting, terminating, and profiling. The main interface shows a summary of the current kernel's performance.

Page: Details **Result:** 1 - 124 - gSum **Add Baseline** **Apply Rules** **Occupancy Calculator** **Copy as Image**

	Result	Time	Cycles	Regs	GPU	SM Frequency	CC	Process
Current	124 - gSum (4, 1, 1)x(4, 1, 1)	3.10 usecond	4,236	16	0 - NVIDIA GeForce RTX 2060	1.36 cycle/nsecond	7.5	[9168] lab3c

GPU Speed Of Light Throughput

High-level overview of the throughput for compute and memory resources of the GPU. For each unit, the throughput reports the achieved percentage of utilization with respect to the theoretical maximum. Breakdowns show the throughput for each individual sub-metric of Compute and Memory to clearly identify the highest contributor. High-level overview of the utilization for compute and memory resources of the GPU presented as a roofline chart.

Metric	Value	Unit
Compute (SM) Throughput [%]	0.09	Duration [usecond]
Memory Throughput [%]	1.47	Elapsed Cycles [cycle]
L1/TEX Cache Throughput [%]	1.33	SM Active Cycles [cycle]
L2 Cache Throughput [%]	1.47	SM Frequency [cycle/nsecond]
DRAM Throughput [%]	0.47	DRAM Frequency [cycle/nsecond]

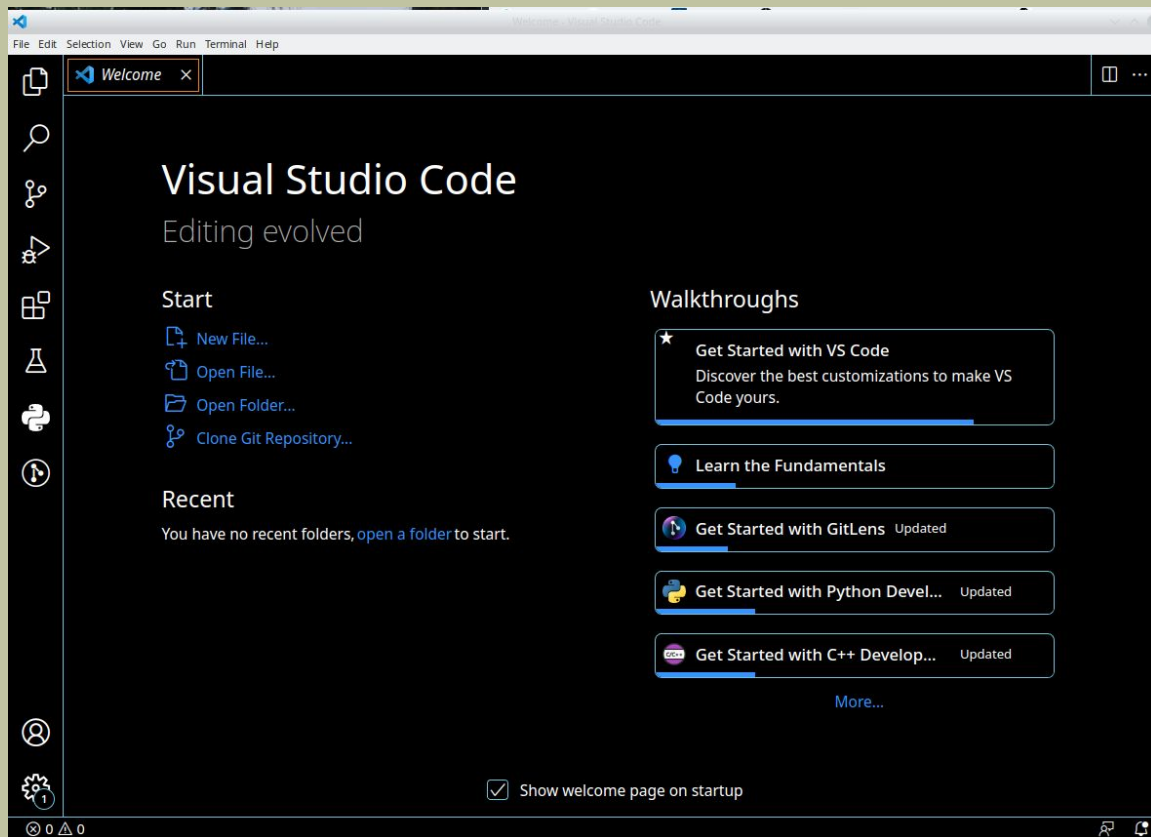
Small Grid This kernel grid is too small to fill the available resources on this device, resulting in only 0.0 full waves across all SMs. Look at [Launch Statistics](#) for more details.

Roofline Analysis The ratio of peak float (fp32) to double (fp64) performance on this device is 32:1. The kernel achieved 0% of this device's fp32 peak performance and 0% of its fp64 peak performance. See the [Kernel Profiling Guide](#) for more details on roofline analysis.

Nsight Visual Studio Code Edition

<https://developer.nvidia.com/nsight-visual-studio-code-edition>

<https://docs.nvidia.com/nsight-visual-studio-code-edition/cuda-debugger/index.html>



File Edit Selection View Go Run Terminal Help

EXPLORER

✓ vsc

> coverage



lab3c.cu

> OUTLINE

> TIMELINE

0 0 Tests ✓ 0 × 0 coverage: progress

> Home > Workshop > VSC

Name	Modified	Size
 coverage	2024-02-11 19:42	
 <u>lab3c.cu</u>	2023-09-23 18:04	



RUN AND DEBUG: RUN

...

Open a file which can be debugged or run.

Run and Debug

To customize Run and Debug create a launch.json file.

Show all automatic debug configurations.

RUN AND DEBUG: RUN

Open a file which can

Run a

To customize Run and launch.json file.

Show all automatic de

Extensions (Ctrl+Shift+X)

Select debugger

CUDA C++ (CUDA-GDB)

CUDA C++ (CUDA-GDBSERVER)

CUDA C++ QNX (CUDA-GDBSERVER)

CMake Debugger

Node.js

Python

Web App (Chrome)

Web App (Edge)

Install extension...

RUN AND DEBUG: RUN

Open a file which can

Run a

To customize Run and
launch.json file.

Show all automatic de

Select debugger

CUDA C++ (CUDA-GDB)

Suggested

CUDA C++ (CUDA-GDBSERVER)

CUDA C++ QNX (CUDA-GDBSERVER)

CMake Debugger

Node.js

Python

Web App (Chrome)

Web App (Edge)

Install extension...

```
> VSC:bach — Konsole
File Edit View Bookmarks Settings Help
(base) malkov@192:~/Workshop/VSC> ls -la
итого 4
drwxr-xr-x 4 malkov users 53 Feb 11 19:48 .
drwxr-xr-x 8 malkov users 93 Feb 11 19:37 ..
drwxr-xr-x 2 malkov users 6 Feb 11 19:42 coverage
-rw-r--r-- 1 malkov users 1084 Sep 23 18:04 lab3c.cu
drwxr-xr-x 2 malkov users 25 Feb 11 19:48 .vscode
(base) malkov@192:~/Workshop/VSC> ls -la .vscode
итого 4
drwxr-xr-x 2 malkov users 25 Feb 11 19:48 .
drwxr-xr-x 4 malkov users 53 Feb 11 19:48 ..
-rw-r--r-- 1 malkov users 534 Feb 11 19:48 launch.json
(base) malkov@192:~/Workshop/VSC> █
```

.vscode > { } launch.json > ...

```
1  {
2      // Use IntelliSense to learn about possible attributes.
3      // Hover to view descriptions of existing attributes.
4      // For more information, visit: https://go.microsoft.com/fwlink/?linkid=829397
5      "version": "0.2.0",
6      "configurations": [
7          {
8              "name": "CUDA C++: Launch",
9              "type": "cuda-gdb",
10             "request": "launch",
11             "program": ""
12         },
13         {
14             "name": "CUDA C++: Attach",
15             "type": "cuda-gdb",
16             "request": "attach"
17         }
18     ]
19 }
```


.vscode > { } launch.json > ...


```
1 {  
2     // Use IntelliSense to learn about possible attributes  
3     // Hover to view descriptions of existing attributes  
4     // For more information, visit: https://go.microsoft.com/fwlink/?linkid=829397  
5     "version": "0.2.0",  
6     "configurations": [  
7         {  
8             "name": "CUDA C++: Launch",  
9             "type": "cuda-gdb",  
10            "request": "launch",  
11            "program": "${workspaceFolder}/lab3c"  
12        }  
13    ]  
14 }
```

>tasks

Tasks: Run Build Task

Ctrl + Shift + B

recently used 

Tasks: Configure Default Build Task 

Tasks: Configure Default Test Task

Tasks: Configure Task

other commands

Tasks: Manage Automatic Tasks

Tasks: Open User Tasks

Tasks: Rerun Last Task

Tasks: Restart Running Task

Tasks: Run Task

Tasks: Run Test Task

Tasks: Show Running Tasks

Tasks: Show Task Log

Tasks: Terminate Task

Select the task to be used as the default build task

Nsight: autostart (localhost)

Nsight: autostart (remote)

Nsight: autostart (secure copy executable binary, remote)

Nsight: autostart (remote QNX)

Nsight: autostart (secure copy cuda-gdbserver binary, remote QNX)

markdownlint: Lint all Markdown files in the workspace with markdownlint



Welcome to GitLens

{ } tasks.json 1 X


{ } launch.json

VSC > .vscode > { } tasks.json > ...


```
1  {
2      "version": "2.0.0",
3      "tasks": [
4          {
5              "type": "shell",
6              "label": "Nsight: autostart (localhost)",
7              "command": "nvcc -g -G lab3c.cu -o lab3c",
8              "problemMatcher": ["$nvcc"],
9              "group": {
10                 "kind": "build",
11                 "isDefault": true
12             }
13         }
14     ]
15 }
```


>build|

Tasks: Configure Default **Build** Task

recently used 

Tasks: Run **Build** Task

Ctrl + Shift + B 

(Internal) **Build** a Target by Name

other commands

npm: Run **Build**

5

6

7

8

"type": "shell",

"label": "Nsight: autostart (1

"command": "make dbg=1",

"problemMatcher": ["\$gcc"]

EXPLORER

...

▼ vsc

▼ .vscode

{} launch.json

{} tasks.json

> coverage

≡ lab3c

lab3c.cu

Makefile

{} launch.json

{} tasks.json 1

lab3c.cu ×

lab3c.cu > main()

27

28 int main(){

29 int N=VECTOR_LENGTH;

30 int *a, *b;

31 int *a_h;

32

33 cudaMalloc((void**)&a, N*sizeof(int));

34 cudaMalloc((void**)&b, N*sizeof(int));



35 a_h=(int*)calloc(N, sizeof(int));

36

37 gInit<<<N/128, 128>>>(a,b);

38 cudaDeviceSynchronize();

39 CUDA_CHECK_RETURN(cudaGetLastError());

R...  CUDA C++  ...

▼ VARIABLES

▼ Local

N: 256

▼ a: 0x7fffc7000000

*a: 0




> b: 0x7fffc7000400


▼ a_h: 0xd25c30









*a_h: 0

> Registers

▼ WATCH

 launch.json

       lab3c.cu 

lab3c.cu > main()

```
27
28 int main(){
29     int N=VECTOR_LENGTH;
30     int *a, *b;
31     int *a_h;
32
33     cudaMalloc((void**)&a, N*sizeof(int));
34     cudaMalloc((void**)&b, N*sizeof(int));
35     a_h=(int*)calloc(N, sizeof(int));
36
37     gInit<<<N/128, 128>>>(a,b);
38     cudaDeviceSynchronize();
39     CUDA_CHECK_RETURN(cudaGetLastError());
```