Indian Institute of Information Technology, Vadodara

Parallel Programming(CS403)

Prof. Reshmi Mitra

Name - Hemant Kumar
ID - 201352026
Collaborator - Dilip Puri(201351014)

Lab 06

Submission Date - October 10, 2016 **Deadline** - Oct 10, 11:59 PM

---

1. In the Callgrind tutorial link and the attached file "Lab6_callgrind.pdf" familarize yourself with the callgrind tool.

2. For the lab exercise, refer to the attached valgrind_eg.c.

```
$ gcc -g -o valgrind_eg valgrind_eg.c -pthread
$ ./valgrind_eg 1
completion time for put phase = 2.758351
0: 0 keys missing
completion time for get phase = 2.270119
This is the same program as used in the previous lab.
valgrind --tool=callgrind --dump-instr=yes --simulate-cache=yes ./val_eg <>
```

   Use the above command on this sample C-code to find the call-graph. Toggle the "Cycle Detection" and "% Relative" menu in the top of kcachegrind to observe to menu output. Using this option, find

   - callee map
   - cost (in terms of cycles) associated with the most expensive function calls.
   - cost (in terms of cycles) associated with pthread_mutex_lock.c and pthread_mutex_unlock.c