NVIDIA

PROFILE: Software

Name of the Role: Software Engineer

Expected GPA requirement: 9+

Eligible departments: CSE

Process involved: Resume-based selection, Combined Technical and HR Interview

Test Details:

- **Interview (approx. 45min):**
  Puzzles, programming questions, and a few questions about OOPS concepts were asked. Questions from the resume, such as about the projects mentioned, were asked. A couple of coding-whiteboard coding questions were asked. Go through a ton of mainstream puzzles and aptitude questions. You need basic coding knowledge to verbatim think about an algorithm and write it. Practice general aptitude puzzles.

  Questions about domains of interest, past internship experiences and college activities (coding clubs, fests) will be asked.

Preparation:
Practice puzzles from GFG, and stay thorough in programming knowledge in C++/ Java or any OOPS language. Learn about Digital systems, computer architecture, C++ and machine learning for interview questions and stay clear about your knowledge on your projects and topics related to that.

Have a strong idea of the company, their background and how that is aligned to your goals and interests. (In my case, I loved supercomputing and Machine learning, and that happened to be a great field to work on GPUs that NVIDIA is a pioneer of.)

JOB DETAILS:

A typical day in this role:

a. My role comes with a heavy dose of imposter syndrome as I am the only college graduate (the rest all have 5+ years experience/PhDs/Masters). Their outlook on projects is way different than newly joined ones. They don't care what technology it is, will get comfortable in a short time and collaborate with experts.
b. The work involved developing Open-source, high-performance, GPU accelerated machine learning libraries in python, C++, and CUDA (checkout https://rapids.ai). Unlike conventional programming, this involves not only high-level algorithmic knowledge and software development knowledge, but also requires a good grasp of systems-level (compiler, computer engineering) details to write GPU-optimized, performant code.
c. My typical day requires me to quickly learn and practice software development skills like DevOps, Git, Coding in C++/python/bash; but also dabble and ramp-up in low-level knowledge like instruction-optimization, GPU architecture, performance profiling, etc. Also, the above work is in effort to develop the best possible, accurate, high-performant and user-friendly ML softwares for data scientists so I frequently end up learning (and relearning) concepts in system design, ML/Data science/Numerical linear algebra.

Expectation vs Reality:

- Expectation:
  - Relaxed work-life balance
  - Slow-paced training process
  - Cool GPUs
  - will have freshers like me in the team
- Reality:
  - Chill job (but the productivity of others is contagious, so I rarely just laze around)
  - No training process. Had to learn concepts on the fly!
    In fact, a big motto of my job culture is, learn the concepts 'only good enough' to solve the immediate problem at hand.

- Yep, I do have access to monster GPUs
- Unfortunately, my team is super small, like only 4 Indians. I happened to be the junior and was always outside my comfort zone!

Your Growth in 1-2 years:
The software teams at NVIDIA are very scarce. Therefore, there is great visibility to upper management. This is a good thing for growth as one can expect collaborations with other countries and even opportunities there. A year of working in this role can force anyone to a level where one owns a particular defined subset of code-base.

Projects and tasks given:
- Implementation of parallelized, optimized versions of existing machine learning algorithms.
- Algorithmic challenges (like the choice of data structures, conjure a parallel algorithm for a machine learning problem)
- Performance challenges (choice of memory, writing compiler-friendly, optimized code that is compiled to hazard-free, high-throughput instructions)
- Learning internal tools for performance profiling.
- DevOps, workflow-related challenges like building, managing, triaging large scale computations on clusters.

Work culture, Employee benefits etc and few points about the company:
The work culture is relaxed, but extremely productive. Everyone loves what they do and is passionate about sharing and helping. In Nvidia, we use a common phrase called ‘speed of light’. The idea is to strive to achieve something until its first principle limit is reached, not to see your competitor for improvement.

- There is no micro-management; no deadlines; no software sprints and hard, time-dependent constraints.
- There is creative pressure but not otherwise.
- Employee benefits include HRA, insurance, stocks (lots of them!), internet benefits, WFH benefits, leaves. Finally, NVIDIA gear store points for GPUs, Nvidia merch, etc.
- The company as a whole is adaptive, bold, and cool. The same mood is preserved in its employees. We have colleagues who are gamers, content creators, Data scientists, etc.

- Most importantly, one does meaningful work and is always looking at what's the next big thing.

Any advice from your side:

Always be explorative and conscious of what excites you; and in that pursuit, keep getting your hands dirty in new things and train your experience. The faster you converge to something you love, the more of a competitive advantage you'll have over your peers, but most importantly, the more you can channel your efforts into something more aligned and meaningful!

- Start doing more leetcode problems FOR FUN! Take your own pace; find satisfaction in solving them.
- Do more hands-on projects as courses (over some internship under a prof. in your early years when the prof. mostly ignores you)
- Learn to document as notes /wiki (using notion, tiddly wiki etc.)
- Learn a good coding editor software (like vim, emacs)
- Learn to Google effectively

Linkedin- https://www.linkedin.com/in/venkataramana-g-90610048/