

NATHAN L. DIEP

nathan.l.diep@gmail.com

www.cs.ucr.edu/~ndiep

OBJECTIVE

A full time Software Engineering or Development Position.

EDUCATION

MASTER OF SCIENCE, COMPUTER SCIENCE - AUGUST 2009

University of California, Riverside
Riverside, CA

BACHELOR OF SCIENCE, COMPUTER SCIENCE - JUNE 2007

University of California, Riverside
Riverside, CA

SKILLS

Programming Languages

C++, Java, Ruby, Python

Software Skills

SQL, HTML, PHP, LaTeX

Software

Microsoft Visual Studio, PostgreSQL, MySQL, Matlab, SVN

Operating Systems

Linux, Windows, Mac OSX

Language Skills

Spoken Languages: English, some Cantonese

EXPERIENCE

MANAGING PARTNER

ELWSOFTWARE.COM — JUNE 2009 - AUGUST 2010

One of the Founding Partners of ELWSOFTWARE.COM, where we created and managed websites for customers. We created and designed websites for customers using HTML, CSS, and PHP. We specialized in SEO and used our own tracking software for the websites. (HTML, CSS, PHP)

TEACHING ASSISTANT

UNIVERSITY OF CALIFORNIA, RIVERSIDE — SEPTEMBER 2007 - JUNE 2009

I worked with various professors to instruct, lead, and manage students in different areas of computer science. Classes taught include: Compiler Design, Introduction to Computer Science, and Introduction to Computing.

CS 151 Compiler Design:

Managed and Taught Lab Sections; graded and prepared solutions for assignments, quizzes, and exams. Provided instruction through the process of creating a compiler to compile a small mini language.

CS 010 & CS 012 Introduction to Computer Science I & II:

Managed and Taught Lab Sections; graded and prepared solutions for assignments, quizzes and exams. Provided instruction on C++ programming and basic data structures, from print statements to classes and linked lists.

CS 008 Introduction to Computing:

Managed and Taught Lab Sections; graded assignments, quizzes and exams. Provided instruction on various computing tools, from Microsoft Office to basic HTML.

PROJECTS

Image Texture Classification using Local Binary Patterns

Using MATLAB I implemented and tested the texture classification algorithm of Local Binary Patterns to classify and group different types of image input textures. I tested the algorithm by inputting various groups of image textures together to see how the algorithm would group the different images together. To see how the algorithm grouped the different textures I would output a dendrogram using euclidian distance measure of the input images for user visibility. This allowed me to test the accuracy and efficiency of the algorithm and test the different methodologies of the Local Binary Patterns algorithm. (MATLAB)

Performance Benchmark of a Super Computer

I performed test and performance benchmarks on the SGI Altix 4700 Super Computer. The SGI Altix 4700 had 64 processing cores and 128 GB of RAM. To benchmark and test the supercomputer I ran the SPLASH 2 benchmark tests using an incremental core count to test the gains received. I used the PAPI API to retrieve the critical performance data from the tests. (BASH Scripting, SPLASH 2, C/C++, PAPI C Interface)

Video Game Creation and Design

I worked in a team of three to design and implement a video game from concept to working prototype. Using the OGRE 3D graphics engine and C++ we programmed a game that allowed a user to fight an AI enemy. We also used full motion capture data to animate the player character and the enemy character. (C++, OGRE 3D)

Ray Tracer

I implemented the Ray Tracer algorithm in C++ to render two spheres (a planet and its moon) with a single light source (the sun). Using the Phong Illumination Model as the lighting model the render drew the two planets and the shadow caused by the moon on the planet as seen by the user. To render the image to the screen OpenGL was used, but the only function used to render the scene was Draw Pixel (turn on a pixel to a desired color). (Open GL, C++)

Travel Agency Database

I implemented a database system for a fictional travel agency. I followed the entire process of creating a relational database, from creating an ER Diagram to implementation of the database system. The database management system used for the implementation was PostgreSQL. The interface to the database was coded in Ruby using SQL queries to access the database. The system allowed the user to manage records of employees, customers, flights, and all the associating attributes. (Ruby, PostgreSQL, SQL)

RELEVANT COURSES

Graduate Courses: Data Mining Techniques, Artificial Intelligence, Database Management, Software Evolution, Design and Analysis of Algorithms, Advanced Computer Networks, Network Routing, Parallel Process Architecture

Undergraduate Courses: Computer Graphics, Video Game Creation and Design, Data Structures and Algorithms, Design and Architecture of Computer Systems, Operating System Design, Compiler Design, Unix System Administration, Embedded Systems

HONORS & ACTIVITIES

Treasurer, Computer Science Graduate Student Association UCR (CompGSA)

Dean's List

Member, Association of Computing Machinery (ACM)

REFERENCES

Available Upon Request