Michael Khayretdinov
Boston, MA / 617-901-94-17 / mkhayret@gmail.com / mkhdv.com
Full-Stack Web Developer

PROGRAMMING LANGUAGES AND TECHNOLOGIES

Java SE8 (currently learning Spring Framework), HTML/CSS, JavaScript ES6+, Oracle SQL, Python, Git, React, Linux, Docker, AWS (S3, EC2, Elastic Beanstalk, RDS, Route 53).

PROJECTS

Game bot - (UMASS Boston, 2020)
As an assignment, I wrote a game playing algorithm for the checkers-like game tournament during the course on Introduction to AI. Used minimax algorithm with iterative deepening and alpha-beta pruning. (Python/NumPy).

Map data visualization for a startup - (UMASS Boston, 2020)
As a project for the Software Engineering Course with a team of 5 students built a visualization tool for the Boston-based startup *Routable.AI*. My responsibility was to deploy GeoJson Layer of deck.gl framework, dynamically displaying routes for a selected fleet of vehicles on the interactive map. (deck.gl, JavaScript, React, Docker, Overleaf).

Java compiler - (UMASS Boston, 2020)
As a series of assignments for the course on Compilers I had to implement a compiler for a subset of Java language j--. On each step of the project I had to reproduce different steps of the language compilation process, from scanning the source language to generating machine code. (Java)

Analysis of crime in Boston - (UMASS Boston, 2019)
As a final project for the course on Probability and Statistics I had to analyze a set of data from the Boston Police Department on several crimes occurring in different areas of Boston. The result was the schedule of patrol cars dispatching based on the probability of crime occurrence in each of the districts. (R, LaTex).

EDUCATION

University of Massachusetts Boston. USA, 2017-2020.
B.S. in Computer Science, Department of Science and Mathematics. Cum laude.
Member of Golden Key Honor Society. Member of Tau Sigma Honor Society.
Fellow at UMASS Venture Development Center Mentorship Program.

COURSES TAKEN

CS 210 - Introduction to Data Structures and Algorithms. The course covered major searching and sorting algorithms and most common data structures as well as introduced graph theory. Was taught in Java.

CS 310 - **Advanced Data Structures and Algorithms**. This course went deeper into the theory of algorithms covering topics like graph theory algorithms, dynamic programming and algorithm analysis. Was taught in Java.

CS 341 - Computer Architecture and Organization. The course covered computer hardware and hardware-level programming concepts. Was taught in Assembly x86 and C.

CS 420 - Introduction to the theory of computation. The course covered the formal basics of theoretical computation covering topics like finite automata, Turing machines, context-free grammars and classes of P and NP problems. Used LaTex and CoqIDE during this course.

CS 430 - Database Management Systems. The fundamental course on database management, introduced to topics like relational algebra, SQL, database design theory, database application development. Taught in Oracle SQL, including SQL/PL and JDBC.

CS 444 - Introduction to Operating Systems. The course covered modern approach to implementation of operating systems, covering topics like memory management, CPU scheduling, device management, process concepts. Was taught in Assembly x86 and C.

CS 450 - **The structure of higher level languages**. This course is an adaptation of SICP course and textbook by MIT, focused on programming languages mechanics, its design and implementation. Was taught in Scheme.

CS 451 - **Compilers**. The course introduced compiler organization and implementation covering topics like algorithms for lexical and syntactical analysis, semantic analysis and code generation. Was taught in Java.

CS 470 - Introduction to Artificial Intelligence. The course introduced main principles of AI filed, covering topics like space-search methods, knowledge representation, artificial neural network, computer vision, and machine evolution. Was taught in Haskell and Python.