BRIDGES assignments grouped by learning outcomes, courses

1 Fundamentals

Covers topics such as printing to console, utilizing in-built functions, file input/output, and other basics of programming not covered by other topics. Classification is aligned with SCS1.

(CS2/DS) 10. Hurricane Tracker Parse CSV files to show where hurricanes are. Stay safe!

(CS1) 16. ImagePuzzle Decypher hidden messages in garbled images and discover PIN codes. This is an assignment in I/Os, loops, and conditions.

(CS1) 27. GameGridBasics Learn the basic of composing an image on the BRIDGES game API.

(CS1) 52. Audio Wave Goal of this assignment is to learn to load audio files and modify it to create new sound waves (averaging, subsampling, etc.)

(CS1) 55. Frequency Player Create a program that plays tones at various frequencies. This is a good assignment for learning about methods, loops, and casting between data types.

(CS2) 65. Reddit Bin reddit posts based on the hour they were posted, and graph it.

(CS1) 72. PixelDrawing This is an assignment using the game grid to save art drawn and to load it back.

2 Analyze runtime efficiency of algorithms related to data structure design

Covers the analysis of varying algorithms and their efficiency, using measured such as BigOh. Classification is aligned with BDSI.

(DS/Alg) 28. BigOhMatters Plotting the runtime of a few hypothetical algorithms to internalize that BigOh is what should matter first.

(DS/Alg) 29. Sorting Bechmark By writting sorting algorithms and benchmarking them, we can understand why it is important to improve the complexity of algorithms. We also understand why it may be good to leave the implementation of well known problems to library implementors.

3 Divide & Conquer Algorithms

Covers the utilization and analysis of divide & conquer algorithms, such as merge sort, quick sort, and convex hull operations. Classification is aligned with BRIDGES.

(DS/Alg) 29. Sorting Bechmark By writting sorting algorithms and benchmarking them, we can understand why it is important to improve the complexity of algorithms. We also understand why it may be good to leave the implementation of well known problems to library implementors.

(DS/Alg) 59. Convex Hull City Data This assignment uses the US city dataset's locations to compute the convex hull using a brute-force algorithm. The computed convex hull will resemble a US map outline.

(DS/Alg) 60. Quadtree City Data This assignment is an example of spatial search queries using a Quadtree data structure. It involves building a quadtree data structure given the locations of a set of US cities and building an adaptive search structure, followed by performing spatial search queries.

4 Select appropriate abstract data types for use in a given application

Covers the interpretation and selection of data structures relevant to a given problem. Classification is aligned with BDSI.

(DS) 8. PQBook Use max-heaps and dictionary to find Shakespeare favorite words!

(Alg) 12. AStarMaze Solves a maze using standard algorithm such as BFS, DFS, Astar. This is a good assignment for stacks and queues, and Shortest Path problems.

5 Greedy Algorithms

Covers the utilization and analysis of greedy algorithms, such as minimum spanning trees, and shortest path algorithms. Classification is aligned with BRIDGES.

(DS/Alg) 9. ShortestPathOSM Build a GPS routing application by accessing maps of your city and computing shortest paths in it. This algorithm teaches about graph algorithms, shortest path algorithms, and priority queues.

(CS2/Alg) 23. Mountain Path Find the simplest path through a mountain by using algorithms. Simply using a heuristics greedy algorithm, or optimally using Dynamic Programming.

(DS/Alg) 44. MST Cities The purpose of this assignment is to implement Prim's Minimum Spanning Tree Algorithm and demonstrate its application using US city data.

6 Conditionals

Covers the utilization of conditional control structures to branch and control the workflow of functions. Classification is aligned with SCS1.

(CS1) 6. GridSquareFill Generate random geometric patterns.

(CS1) 7. GridLyrics Visualizing repetition in the lyrics of song makes their structure appear. This assignment teaches loops, condition, and basic string processing while having a data visualization flair to it.

(CS1/CS2) 11. 2048 Game Build the popular game 2048.

(CS2) 13. Infinite Runner Run, jump over bombs, and grab apples!

(CS2/Alg) 14. Spreading Fire Simulate the spreading of fire in different conditions. This is a great assignment to learn about simulation, looping, conditions, 2D arrays.

(CS1/CS2) 15. FallingSand Simulate the physics of sand and water. This is an interactive assignment using loops, conditions, and arrays.

(CS1) 16. ImagePuzzle Decypher hidden messages in garbled images and discover PIN codes. This is an assignment in I/Os, loops, and conditions.

(CS1) 19. Bugstomp See a bug and stomp on it. It is some kind of a whack-a-mole clone. This is good to teach basic tests and condition.

(CS1/CS2) 21. RaceCar Make a racing game in Bridges with ever shrinking tracks. How long can you race? This assignment teaches about loops, 1D arrays, and conditions.

(CS1/CS2) 22. Snake Build the classic Snake game. This game teaches about loops, conditions, and queues.

(CS1) 25. Patterns Create colorful graphical patterns. This is a good assignment for loops, and conditions.

(CS1/CS2) 35. Connect Four Create Connect Four and play against your friends!

(CS1/CS2) 36. Pong Implement Pong and play against your friends!

(CS1) 40. Image Processing Use simple Color and Image Processing Operations using the Bridges ColorGrid

(CS1/CS2) 41. Space Invadors Make a game similar to Space Invaders. This assignment teaches about loops, conditions, and 2D arrays.

(CS2/Alg) 64. Voronoi Diagram The Voronoi diagram is a partition of a plane into regions such that all points in a region are closer to a Voronoi site (or object) than any other site.

(CS2/DS) 69. Snake Queues The classic game of Snake. Move a "snake" (line of sprites) along a 2D grid attempting to run over a randomly placed objects (apple, bombs) to help it grow or shrink. If the snake runs into itself the game is over and the player has lost. The objective of the game is to make the snake as big as possible.

(CS1/CS2) 70. WordleBot The assignment is a variation of the popular Wordle game, but you are writing the game's logic to find the solution given random guesses from a dictionary of words.

(CS1) 71. Pi Estimation N/A

7 Compare data structure tradeoffs to select the appropriate implementation for an abstract data type

Covers the comparison and subsequent analysis of determining how appropriate multiple data structures are at implementing a problem. Classification is aligned with BDSI.

(CS1/CS2) 22. Snake Build the classic Snake game. This game teaches about loops, conditions, and queues.

(CS2/DS) 69. Snake Queues The classic game of Snake. Move a "snake" (line of sprites) along a 2D grid attempting to run over a randomly placed objects (apple, bombs) to help it grow or shrink. If the snake runs into itself the game is over and the player has lost. The objective of the game is to make the snake as big as possible.

8 Brute Force Algorithms

Covers the utilization and analysis of brute force algorithms, such as convex hull and voronoi diagram. Classification is aligned with BRIDGES.

(DS/Alg) 59. Convex Hull City Data This assignment uses the US city dataset's locations to compute the convex hull using a brute-force algorithm. The computed convex hull will resemble a US map outline.

(CS2/Alg) 64. Voronoi Diagram The Voronoi diagram is a partition of a plane into regions such that all points in a region are closer to a Voronoi site (or object) than any other site.

9 Design and modify data structures capable of insertion deletion search and related operations

Covers the implementation and modification of data structures which can perform these operations. Classification is aligned with BDSI.

(DS) 1. ListIMDB Access and visualize remote data through Bridges.

(DS) 3. Graph Bacon Number Compute the Bacon number of famous actors on IMDB's actor movie graph. This is a good basic graph assignment requiring to implement BFS.

(DS) 5. BST Earthquake Illustrates how to access Earthquake data and display it using a binary search tree.

(DS) 8. PQBook Use max-heaps and dictionary to find Shakespeare favorite words!

(DS/Alg) 60. Quadtree City Data This assignment is an example of spatial search queries using a Quadtree data structure. It involves building a quadtree data structure given the locations of a set of US cities and building an adaptive search structure, followed by performing spatial search queries.

10 Graphs & Traversal Algorithms

Covers the utilization and analysis of graphs and their traversal algorithms, such as breadth first search and depth first search. Classification is aligned with BRIDGES.

(DS) 3. Graph Bacon Number Compute the Bacon number of famous actors on IMDB's actor movie graph. This is a good basic graph assignment requiring to implement BFS.

(DS/Alg) 4. GraphEQ Visualize in real time earthquakes and derive proximity graphs. This is a basic Graph construction assignment. Optionally, one can highlight connected components with different colors.

(DS) 8. PQBook Use max-heaps and dictionary to find Shakespeare favorite words!

(DS/Alg) 9. ShortestPathOSM Build a GPS routing application by accessing maps of your city and computing shortest paths in it. This algorithm teaches about graph algorithms, shortest path algorithms, and priority queues.

(CS2/DS) 10. Hurricane Tracker Parse CSV files to show where hurricanes are. Stay safe!

(Alg) 12. AStarMaze Solves a maze using standard algorithm such as BFS, DFS, Astar. This is a good assignment for stacks and queues, and Shortest Path problems.

(DS/Alg) 30. TemporalBaconNumber See the evolution of actors by tracking their Bacon number over time. Compute BFS on different graphs and study the complexity of the analysis.

(DS) 31. TemporalPageRank Was Kevin Bacon always the center of Hollywood? Who was Kevin Bacon before Kevin Bacon? Answer these questions by computing PageRank on a Movie-Actor graph from 1910 to today. Study the complexity of PageRank on large graphs.

(Alg) 39. Book Distance Textual analyis of books using the Project Gutenberg book collection

(DS/Alg) 43. Spatial Indexing In this assignment we will use Open Street map data to retrieve a set of locations from a specified section of a city or region. The goal is to efficiently compute the closest point given a 2D query location.

(DS/Alg) 44. MST Cities The purpose of this assignment is to implement Prim's Minimum Spanning Tree Algorithm and demonstrate its application using US city data.

11 Iterative Structures

Covers the utilization of iteratives, such as loops, lists, arrays, and other structures. Classification is aligned with BRIDGES.

(CS1) 6. GridSquareFill Generate random geometric patterns.

(CS1) 7. GridLyrics Visualizing repetition in the lyrics of song makes their structure appear. This assignment teaches loops, condition, and basic string processing while having a data visualization flair to it.

(CS1/CS2) 11. 2048 Game Build the popular game 2048.

(CS2) 13. Infinite Runner Run, jump over bombs, and grab apples!

(CS2/Alg) 14. Spreading Fire Simulate the spreading of fire in different conditions. This is a great assignment to learn about simulation, looping, conditions, 2D arrays.

(CS1/CS2) 15. FallingSand Simulate the physics of sand and water. This is an interactive assignment using loops, conditions, and arrays.

(CS1) 16. ImagePuzzle Decypher hidden messages in garbled images and discover PIN codes. This is an assignment in I/Os, loops, and conditions.

(CS1) 19. Bugstomp See a bug and stomp on it. It is some kind of a whack-a-mole clone. This is good to teach basic tests and condition.

(CS1/CS2) 20. Minesweeper Build the classic Minesweeper game. This game teaches about loops, tests, 2D arrays, and recursion.

(CS1/CS2) 21. RaceCar Make a racing game in Bridges with ever shrinking tracks. How long can you race? This assignment teaches about loops, 1D arrays, and conditions.

(CS1/CS2) 22. Snake Build the classic Snake game. This game teaches about loops, conditions, and queues.

(CS2/Alg) 23. Mountain Path Find the simplest path through a mountain by using algorithms. Simply using a heuristics greedy algorithm, or optimally using Dynamic Programming.

(CS1) 25. Patterns Create colorful graphical patterns. This is a good assignment for loops, and conditions.

(CS1) 33. Audio Mixing Learn about audio mixing using Bridges AudioClip. Mix two audio sources together into one. Also mix two audio sources by fading one into the other.

(CS1/CS2) 35. Connect Four Create Connect Four and play against your friends!

(CS1) 40. Image Processing Use simple Color and Image Processing Operations using the Bridges ColorGrid

(CS1/CS2) 41. Space Invadors Make a game similar to Space Invaders. This assignment teaches about loops, conditions, and 2D arrays.

(CS1) 52. Audio Wave Goal of this assignment is to learn to load audio files and modify it to create new sound waves (averaging, subsampling, etc.)

(CS1) 55. Frequency Player Create a program that plays tones at various frequencies. This is a good assignment for learning about methods, loops, and casting between data types.

(CS2/Alg) 56. 2D Indexing This assignment is part of a larger assignment [insert link here] on determining the closest point to a given point within a 2D map, eg., finding the nearest restaurant from a given location. It initiates a search from within a 2D grid and enumerates the cells layer by layer outwards from the start cell.

(CS2/Alg) 64. Voronoi Diagram The Voronoi diagram is a partition of a plane into regions such that all points in a region are closer to a Voronoi site (or object) than any other site.

(CS2) 65. Reddit Bin reddit posts based on the hour they were posted, and graph it.

(CS2/DS) 69. Snake Queues The classic game of Snake. Move a "snake" (line of sprites) along a 2D grid attempting to run over a randomly placed objects (apple, bombs) to help it grow or shrink. If the snake runs into itself the game is over and the player has lost. The objective of the game is to make the snake as big as possible.

(CS1) 72. PixelDrawing This is an assignment using the game grid to save art drawn and to load it back.

12 Functions

Covers the utilization of functions and methods to complete tasks, such as controlling objects or manipulating data. Classification is aligned with BRIDGES.

(CS2) 13. Infinite Runner Run, jump over bombs, and grab apples!

(CS1/CS2) 70. WordleBot The assignment is a variation of the popular Wordle game, but you are writing the game's logic to find the solution given random guesses from a dictionary of words.

13 Recursion

Covers the concept of recursion, both mathematically and implemented to solve recursive problems. Classification is aligned with SCS1.

(CS1/CS2) 20. Minesweeper Build the classic Minesweeper game. This game teaches about loops, tests, 2D arrays, and recursion.

(DS/Alg) 24. Image Compression Representing/Compressing 2D Images Using K-D Trees

(CS2/DS) 26. TowersOfHanoi A classic puzzle involving ancient Vietnamese temples and recursion.

(DS/Alg) 59. Convex Hull City Data This assignment uses the US city dataset's locations to compute the convex hull using a brute-force algorithm. The computed convex hull will resemble a US map outline.

(DS/Alg) 60. Quadtree City Data This assignment is an example of spatial search queries using a Quadtree data structure. It involves building a quadtree data structure given the locations of a set of US cities and building an adaptive search structure, followed by performing spatial search queries.

(CS2/Alg) 74. Fractals N/A

(CS2/Alg) 75. Koch Curve Understand how to use recursion to create complex shapes like the koch curve and koch snowflake. Use AffineTransform to rotate in Java.

(CS2/Alg) 76. Zcurve Z-order curve is a type of space filling curve that can be generated very easily using recursion For more info on Z-order check out the wikipedia page

14 Object Oriented Basics

Covers the basic concepts of OOP, including objects/classes, super classes, etc. Classification is aligned with SCS1.

(CS1) 19. Bugstomp See a bug and stomp on it. It is some kind of a whack-a-mole clone. This is good to teach basic tests and condition.

(CS1/CS2) 36. Pong Implement Pong and play against your friends!