Mikayla Timm

12/06/2015

COP4534

Optimal Parens Functional Decomposition

|  |  |  |
| --- | --- | --- |
| File | Functions | Purpose |
| Matrix.h | None | Function and struct declarations for Matrix.c |
| Matrix.c | 1. MatrixChainP runPart1();  2. void MatrixChainOrder(MatrixChainP mcp);  3. void PrintOptimalParens(int \*charCounter, MatrixChainP mcp, int i, int j);  4. void inToPost(int \*charCounter, MatrixChainP mcp, int i, int j);  5. void matrixMultiplication(MatrixChainP mcp, MatrixP matrix1, MatrixP matrix2, MatrixP result);  6. void display(MatrixP matrixToShow);  7. void evaluatePostfix(MatrixChainP mcp);  8. void pushStack(StackP stack, int value);  9. int popStack(StackP stack); | 1. Runs part 1 of the project: finding optimal paren  2. Dynamic programming solution to finding the optimal parenthesization  3. prints and saves the optimal parenthesization  4. Converts the infix expression into postfix  5. Multiplies two matrices together and saves the result, keeping track of number of scalar multiplications  6.displays the values in the matrix  7. Evaluates the multiplication indicated by the postfix expression and saves the result into the last matrix in the matrix array.  8.Push an index off the stack we're using for evaluating the expression  9. Pop an index off the stack we're using for evaluating the expression |
| FileIO.h | None | Function declarations for FileIO.c |
| FileIO.c | 1. void getMatrixDim(MatrixChainP mcp);  2. void getMatrixData(MatrixChainP mcp); | 1. Opens MatrixDimensions.txt for reading. Gets the number of dimensions and the actual dimensions  2. Read in all of the matrix data from matrixData.txt |
| OptimalParensTest.c | Main.c | Runs the program |