

Exhaustive Crypto Problem Solver with Elaboration

Name: Miguel Rodriguez

Course: CSC366

The main purpose of the Exhaustive Crypto Problem Solver program is to generate either random crypto problems from the range of 0 to 15 with order 2 to 5 or to solve user non-randomly specified crypto problems and display their solutions. Users can also solve many random crypto problems and display their solutions at once.

The program starts by consulting the global variable ADT and combosets.pro files. Thereafter, it establishes the crypto program parameters where it generates 5 random numbers from the range of 0 to 15 and a goal from within the same range. Moreover, it binds the numbers to variables and store them in the knowledge base using the function “addCryptoProblemToKnowledgeBase” where they can be retrieve and display using the function “displayProblem”. The problems are exhaustively solve using the four basic arithmetic operators but when it comes to subtraction and division the numbers are switch from high to low, having a subtraction and division from low to high and high to low. The program also checks for division by zero. Users can generate single random crypto problems by typing “solve(random).” which generates a random crypto problem, displays it, solves it decomposionally, and displays the solution. Furthermore, users can specify the crypto problem to be solve by typing “solve (numbers (N1, N2, N3, N4, N5), goal(G))” where the “establishSpecificCryptoProblem” function is used to store the problem to the knowledge base, then it is solved, and then display to the user with its solution. The function “solveProblemDecompositionally” gets the crypto problem from the knowledge base, solves it, and adds the solution to the knowledge base using the function

“addCryptoSolutionToKB(Expression)”, and then prints it to the console using the function “displaySolution” or prints “No solution to this one!” if there is no solution to the problem.

Finally, the “demo (N)” function generates random crypto problems within the range of 0 to 15 for as many times as the user specifies in its parameters by using recursion, and displays them with their corresponding solutions.