MIDDLE EAST TECHNICAL UNIVERSITY, NORTHERN CYPRUS CAMPUS

CNG438 Information and Network Security: Practical Assignment 2

Date handed out: See ODTUClass

Date submission due: See ODTUClass

DES Implementation with Operation Mode

Purpose

The objective of this assignment is to familiarize undergraduate students with the Data Encryption Standard (DES) algorithm and its mode of operations. Students will implement the DES algorithm and an operation mode in a programming language of their choice and gain hands-on experience in encrypting and decrypting data using these cryptographic techniques.

Tasks

Implementation of DES

- Implement the DES algorithm in a programming language of their choice (e.g., C#). **If you choose C#, you will get bonus points.**
- Test your implementation with sample inputs and verify the correctness of your encryption and decryption functions.

The DES algorithm and the implementation details such as S-Box structure and mapping structure is given in the class. If you have any trouble finding them, please see the instructor.

Implementation of Operation Mode

- You will only implement one of the modes. You will select which mode you will implement by following the criteria given in the table below.

Operation Mode that will be implemented	**Criteria (by the students whose student number ends with the following digit(s)).**
ECB	0, 2
CBC	1, 8
CFB	3
OFB	4, 5
CTR	7, 9

- Extend your DES implementation to support the mode encryption and decryption.

Assessment Criteria

- If your program does not compile, then you will get 0.
- Correctness and functionality of DES and CBC mode implementations.

- Clarity and organization of code documentation.
- Adherence to submission guidelines and deadlines.

Grading

Your program will be graded as follows:

Criteria	**Mark (100)**	**Remarks**
DES Implementation	40	Assessing correctness of DES encryption/decryption, Efficiency of implementation (speed, memory), Clarity of code documentation and comments.
Mode Implementation	40	Evaluating correctness of CBC mode encryption/decryption, Proper handling of Initialization Vector (IV), Integration with DES implementation.
Testing	10	Evaluating the accuracy of testing against provided vectors.
Reporting	5	Assessing clarity and organization of implementation details, Adherence to submission guidelines.
Overall Presentation	5	Evaluating the quality of presentation including professionalism in formatting, language usage, and visual appeal.

Submission

You need to submit a ZIP file (firstname_lastname.zip, e.g., okan_topcu.zip) including the following:

1. Your source codes.
2. Your executable if needed.
3. *Readme.txt* file for your source code. This should include a short description of your program and **it should explain how to compile and run your code**, please include your name, surname, and student id at the top.
4. A short video showing the execution demo. You may provide a link to an outside site (e.g., YouTube)

To submit your assignment, simply select the appropriate assignment link from the ODTUCLASS page. Upload your zip file and click submit (clicking send is not enough). Please make sure all source files are included in your zip file when submitted. Only your final submission will be graded. Remember there is no late submission for this assignment.

Academic Honesty

In the context of academic assignments, it is imperative to underscore the importance of hands-on learning and the mastery of coding skills. The utilization of generative AI, while potentially beneficial, should primarily serve as a tool for educational enhancement rather than a shortcut to circumvent the learning process.

Throughout the assignment, students are expected to engage directly with the implementation of cryptographic algorithms, such as DES and CBC mode, through coding exercises. The ultimate goal of these assignments is not merely the production of correct outputs but the acquisition of knowledge and proficiency in cryptographic techniques. In this pursuit, the use of generative AI should be approached with caution, ensuring that its application aligns with the principles of academic integrity and the educational objectives of the assignment.

While AI-generated content can provide insights and assistance, it should not overshadow or replace the active learning process inherent in coding and problem-solving.

In completing this assignment, it is imperative to uphold the principles of academic integrity and honesty. Collaboration can be a valuable learning tool, but it is essential to distinguish between collaboration and unauthorized collaboration, which includes sharing code with peers or seeking solutions from external sources without acknowledgment. Cheating undermines the educational process and diminishes the value of the assignment, both for yourself and for your peers. It erodes trust within the academic community and devalues the skills and knowledge gained through individual effort and exploration.

While discussions with peers about concepts and strategies are encouraged, the code you submit must be your own original work. Sharing code, whether partially or in its entirety, with peers is considered a violation of academic integrity and will not be tolerated.

You need to submit your own solution. Your source code will be investigated for neatness and cheating. All the programming assignments will be required to be implemented individually and any code sharing will be considered as cheating. Please note that if you are caught cheating, you can get zero from this assignment and also from all the others. See the course syllabus for further details and clarification.