Scientific and Practical Cyber Security Journal (SPCSJ) 8(2): 40 – 50 ISSN 2587-4667 Scientific Cyber Security Association (SCSA)

MATHEMATICAL MODELING: ALGORITHMIC STRUCTURES OF DOD COMMON ACCESS CARDS

Charde'Lyce Edwards BSCS
Full Sail University, Computer Science

ABSTRACT: This study presents a comprehensive examination of the mathematical complexities inherent in the department of defense (dod) common access cards (cac). It delves into the historical progression of the program, tracing its evolution alongside advancements in mathematical modeling. With a specific focus on the algorithmic generation of the bar-code, this paper unveils the equation orchestrating this crucial security feature, vital for maintaining the integrity and security of dod cac cards. Through the unraveling of this mathematical tapestry, the research offers compelling insights into the intricate interplay between mathematics and modern security technology. By seamlessly blending historical context with a detailed exploration of algorithmic intricacies, this study provides a thorough and captivating analysis of this pivotal aspect of security technology. Moreover, it sets the stage for understanding the profound implications of mathematical modeling on security measures, underscoring the critical role of mathematics in shaping modern security practices.

KEYWORDS: *Department of Defense (DOD), Common Access Cards (CAC), Mathematical Modeling, Algorithmic Structures, Security Technology*

1. INTRODUCTION

Bar-codes stand as the quintessential symbol of technology's seamless integration into our daily routines, offering efficiency and convenience in various domains, from retail transactions to inventory management. As succinctly noted, "Bar-codes are the most common type of encoding, a feature available in almost all ID card software. They are quick and affordable to print because they do not require special ribbons, cards, encoding modules, or advanced software." (Barcode Encoding - Introduction to ID Card Software - Learning Center — AlphaCard, n.d.).

However, in the realm of bar-code technology, it becomes evident that its sig- significance transcends beyond commercial applications, particularly within secure environments such as the Department of Defense (DOD), where Common Access Cards (CAC) serve as essential tools for authentication and access control. The use of bar-codes in security applications underscores the adaptability and versatility of this technology in safeguarding sensitive information and controlling access to restricted areas.

This research embarks on a comprehensive exploration of the mathematical foundations that support the algorithmic structures of DOD Common Access Cards. Unlike previous approaches, this study introduces an updated method for defining the math- mathematical equation that governs the generation of PDF417 bar-codes. By tracing the historical evolution of bar-code technology and advancements in mathematical modeling, the symbiotic relationship between mathematics and modern security technology is emphasized, highlighting the significance of these insights for ensuring the integrity and security of access credentials. Moreover, it sets the stage for understanding the profound implications of mathematical modeling on security measures, underscoring the critical role of mathematics in shaping modern security practices.

1.1 BACKGROUND

1.1.1 HISTORY AND DEVELOPMENT OF PDF417 BAR-CODES

The emergence of the bar-code dates back to the 1940s when Bernard Silver and Norman Joseph Woodland envisioned a system to streamline supermarket checkout processes. Their vision culminated in 1974 with the development of the Universal Product Code (UPC) by a team led by George Laurer, marking a pivotal moment in bar-code history. Initially conceived as a one-dimensional encoding system, the UPC revolutionized retail operations, ushering in an era of faster and more accurate product scanning. As bar-codes permeated every facet of modern life, their significance transcended beyond commercial transactions to encompass security and identification. However, the limitations inherent in one-dimensional bar-codes soon prompted the exploration of more advanced encoding technologies. This pursuit led to the inception of two-dimensional (2D) bar-codes, which, grounded in mathematical principles, naturally evolved to tackle the intricacies of ensuring seamless data retrieval across diverse applications.

Among the most notable formats is PDF417, introduced in the 1990s by Dr. Ynjiun P. Wang, boasting enhanced data storage capabilities within a compact footprint. PDF417 bar-codes revolutionized bar-code applications by incorporating mathematical algorithms to generate spacing and output, facilitating secure identification and com- comprehensive data storage across various sectors. The discovery of this secure method prompted institutions such as the Department of Defense (DOD) to leverage the versatility of bar-codes, particularly PDF417, by implementing Common Access Cards (CAC), thereby enhancing authentication and access control measures within secure environments.

Having explored the historical and developmental journey of PDF417 bar-codes, it's essential to delve deeper into their structural components and encoding principles. From its humble beginnings in supermarket checkout lines to the applications in security technology, the bar-code has emerged as an indispensable tool in the digital landscape of the modern era. (The Science Behind Barcode Decoding Algorithms - FasterCapital, n.d.)

1.1.2 DESCRIPTION OF PDF417 BAR-CODE STRUCTURE AND COMPONENTS

The PDF417 bar-code consists of multiple rows of linear bar-code symbols stacked on top of each other. Each symbol represents a codeword that encodes a specific set of data and is structured into three main components:

1. **Start and Stop Patterns**: The PDF417 bar-code begins with a start pattern and ends with a stop pattern, which provide the boundaries of the bar-code. These patterns help scanners identify the beginning and end of the bar-code.
2. **Codewords**: The data encoded in the PDF417 bar-code is divided into codewords. Each codeword represents a character or a set of characters, depending on the encoding mode used. PDF417 supports multiple encoding modes, including text, numeric, byte, and byte compaction.
3. Error Correction Codewords: PDF417 includes built-in error correction capabilities to ensure data integrity even if the bar-code is partially damaged or obscured. Error correction codewords are added to the bar-code to detect and correct errors during scanning. The level of error correction can be adjusted to balance data capacity and redundancy.(CHIPS Articles: Safeguarding Your Common Access Cards and Military Identification Cards, n.d.)

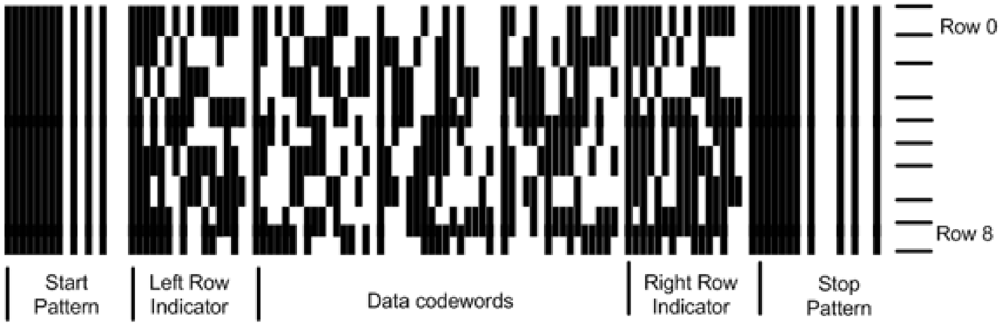

Fig.1. *Anatomy of PDF417 bar-code*

Additionally, the PDF417 bar-code structure is flexible and can accommodate various data formats and sizes. It supports up to 90 different symbols, each consisting of 17 modules (black and white bars) in width and up to 30 rows in height.

1. **Quiet Zones**: PDF417 bar-codes require quiet zones, also known as margins, on all sides to ensure accurate scanning. These quiet zones are empty spaces that provide separation between the bar-code and any surrounding text or graphics.
2. **Data Columns**: PDF417 bar-codes are divided into data columns, each containing a set number of codewords. The number of data columns determines the bar-code's width and affects its overall data capacity.
3. **Rows**: PDF417 bar-codes consist of multiple rows of codewords stacked vertically. The number of rows depends on factors such as the amount of data encoded and the desired bar-code size.
4. **Structured Append**: PDF417 supports structured append, allowing multiple PDF417 symbols to be linked together to encode larger datasets. This feature enables efficient data management and retrieval.(2 Overview - PDF417 Fonts Encoder 5 User Manual, n.d.)

1.1.3 ENCODING PRINCIPLES AND SPECIFICATIONS

Encoding principles are fundamental to PDF417's versatility, enabling the representation of vast amounts of data within a limited space. Unlike traditional linear bar-codes, PDF417 transcends the confines of a single dimension, harnessing the power of two- dimensional encoding. As explored by DENSO WAVE, ”..information can be encoded in both the transverse and the longitudinal directions (two dimensions).' This bidirectional encoding capability not only enhances data capacity but also facilitates robust error correction, ensuring the integrity and reliability of information retrieval.” (What Is a 2D Code? Technical Information of Automatic Identification DENSO WAVE, n.d.) The integration of both transverse and longitudinal encoding epitomizes the sophistication underlying PDF417's encoding principles, emphasizing its indispensable role in modern security and data management contexts

2 MATHEMATICAL FOUNDATIONS

2.1 MATHEMATICAL MODELING OF PDF417 BAR-CODE GENERATION

Mathematical modeling of PDF417 bar-code generation entails a meticulous endeavor to ensure precise data encoding within the symbol's complex framework. The PDF417 symbol can encompass varying numbers of rows, ranging from as few as 3 to as many as 90. Each row adopts characters from a designated cluster for data encoding, with cluster assignment repeating systematically every third row. This organized allocation of clusters to rows is governed by a formula, ensuring efficient decoding strategies and enabling scanners to decipher PDF417 symbols seamlessly, even across multiple rows:

Cluster number = ((row number − 1) mod 3) × 3

facilitates efficient decoding strategies, enabling scanners to decipher PDF417 symbols seamlessly, even across multiple rows.

The structural composition of rows adheres to a distinct pattern, commencing with a left-row indicator (L) followed by a sequence of data region characters, with the most significant characters positioned adjacent to the left-row indicator. Subsequently, each row concludes with a right-row indicator (R). These row indicators, symbolized characters located near the start and stop patterns, encode critical information such as the row number, total row count (ranging from 3 to 90), data region column count (ranging from 1 to 30), and error correction level (ranging from 0 to 8). This meticulous structuring of rows and clusters within the PDF417 symbol ensures efficient data encoding and robust bar-code integrity, facilitating accurate and reliable data retrieval across various applications. (PDF417 Specification for Barcode Symbology, n.d.).

After establishing the fundamental principles of PDF417 bar-code generation, the mathematical foundations provide a comprehensive framework for understanding the intricate algorithms and equations governing their structure and encoding. From calculating row height to implementing error correction mechanisms, each aspect con- tributes to the robustness and reliability of PDF417 bar-codes across various security and data management applications.

2.1.1 EQUATIONS FOR CALCULATING ROW HEIGHT

The fundamental unit governing the bar-code's structure is the module, dictating crucial aspects of its design. Central to this is the individual row height, intricately tied to the module height, ensuring uniformity and precision within the symbol. Complementing this, the width of a module, commonly denoted as the X dimension, further defines the symbol's spatial dimensions. However, it is the ratio of module height to width, known as the y-height, that holds particular significance.

$$W = (17C + 69)X + 2Q$$

$$H = RY + 2Q$$

where:

H = Height of symbol
W = Width of symbol
C = Number of columns in the data region
R = Number of rows
X = X-dimension of the symbol
Y = Row height
Q = Size of Quiet Zone (minimum 2X)

Fig.2. *Width and Height Equations (PDF417 Specification for Barcode Symbology, n.d.)*

With industry standards advocating for a minimum y-height of 3.0, emphasis is placed on maintaining adequate proportions for enhanced decodability. This meticulous attention to detail underscores the commitment of the symbology to optimal readability and reliability, ensuring that PDF417 symbols fulfill their role effectively across various applications. (2 Overview - PDF417 Fonts Encoder 5 User Manual, n.d.)

2.1.2 OVERVIEW OF BASE 929 ENCODING SCHEME

The BASE 929 encoding scheme, integral to PDF417 bar-code generation, employs a meticulous process to compute error correction codewords, ensuring data integrity and reliability. These codewords are derived as the complement of coefficients obtained by dividing the symbol data polynomial $d(x)$ by the generator polynomial $g(x)$, multiplied by x^k, within the Galois Field GF(929). Negative values within this field are treated with the complement of their absolute values, ensuring consistency in encoding. The generator polynomial for k error correction codewords is defined as

$$g(x) = (x - 3)(x - 3^2) \cdots (x - 3^k)$$

serving as the cornerstone of the encoding process. (PDF417 Specification for Barcode Symbology, n.d.)

Among all physical parts of a PDF-417, the data codewords are used for encoding the message data, which could be numbers, letters, or other symbols. PDF417 uses Reed–Solomon error correction. PDF417 bar-code symbology adopts a base 929 mode, where each codeword represents a numeric digit ranging from 0 to 928. Among these codewords, 900 are designated for information data encoding, while the remaining 29 serve specific functions within the bar-code. PDF417 supports three distinct encoding modes—text, byte, and numeric—which can be combined as needed. In the text mode, each codeword signifies 1 or 2 characters, while in the byte mode, groups of 5 codewords represent 6 bytes. Numeric encoding mode utilizes groups of 15 codewords to represent 44 decimal digits, ensuring flexibility and efficiency in data representation within PDF417 bar-codes. (VB.NET PDF417 Generator — Generate, Draw PDF417 Barcode Image In VB.NET Applications With Valid Data Input, n.d.)

2.2 ALGORITHM DESIGN

The evolution of the PDF417 bar-code generation algorithm reflects a journey marked by refinement and abstraction, drawing inspiration from the principles of Set Theory. Originally, the algorithm involved intricate calculations to determine row height (Y) and cluster number (K) for each codeword, encapsulated in separate equations:

1. **Calculate Row Height (Y) in terms of X (minimum width of a vertical bar):**

$Y \geq 3X$

This ensures that the height of each row (Y) is at least three times the width of a vertical bar (X), providing sufficient space for encoding data.

2. **Calculate Cluster Number (K) for Each Codeword: Equation 1:**

$$K = b_1 - b_2 + b_3 - b_4 + 9 \mod 9$$

Equation 2:

$$K = E_1 - E_2 + E_5 - E_6 + 9 \mod 9$$

By following these steps, the algorithm generates each row of the PDF417 bar- code, incorporating the necessary components such as quiet zones, start and stop patterns, and data codewords, resulting in the complete bar-code symbol.

However, guided by the pursuit of elegance and clarity, these calculations under- went a transformation. Through a process of consolidation, they were unified into a single expression, symbolized by the union operator, reflecting the essence of Set Theory.

$$\text{PDF417\_Barcode} = \bigcup_{i=1}^{N_{\text{rows}}} \text{Row}_i$$

Fig.3. *Proposed Equation*

This abstraction not only streamlined the algorithms for row height and the cluster but also imbued it with a sense of cohesion and clarity, echoing the rigorous standards of mathematical modeling.

2.2.1 STEP-BY-STEP EXPLANATION OF THE ALGORITHM

The process of generating a PDF417 bar-code can be summarized into several key steps. At its core, the PDF417 bar-code comprises multiple rows, each containing essential components for encoding data. The total number of rows, denoted as N_{rows}, encapsulates the entire bar-code symbol. Within each row, represented as Row_i, various components are arranged to ensure accurate encoding and decoding. These components include the quiet zone, start and stop patterns, row left and right codewords, and the data codewords themselves. By structuring the bar-code in this manner, the algorithm orchestrates the generation of a complete PDF417 bar-code symbol, ready for scanning and interpretation.

2.2.2 CONSIDERATIONS FOR PARAMETER SELECTION AND OPTIMIZATION

However, there are several considerations that should be noted when utilizing this formula:

1. **Union of Rows**: The formulation of the PDF417 bar-code algorithm involves the union of multiple rows, each containing essential components such as quiet zones, start and stop patterns, data codewords, and row left and right code- words. The selection of parameters for each row, including the size of quiet zones and the placement of codewords, directly impacts the overall bar-code quality and readability.

[2]These equations determine the cluster number (K) for each codeword in the PDF417 bar-code. The values b_1, b_2, b_3, b_4 and E_1, E_2, E_5, E_6 are derived from the encoded data, and the modulo operation ensures that the result remains within a valid range.

2. **Row Height**: Optimizing the height of each row ensures sufficient space for encoding data and maintaining bar-code readability. The row height should be carefully chosen to accommodate the required number of data codewords while adhering to industry standards and best practices.
3. **Start and Stop Patterns**: The design and placement of start and stop patterns play a crucial role in bar-code scanning and decoding. Parameters such as pat- tern size, spacing, and alignment should be optimized to facilitate reliable bar-code recognition and decoding.
4. **Quiet Zones**: Adequate quiet zones between rows and around the bar-code symbol are essential to minimize interference and ensure accurate scanning. The size of quiet zones should be carefully determined based on scanning device requirements and environmental conditions.
5. **Data Codewords**: The selection of parameters for data codewords, including encoding mode, error correction level, and data compression, influences bar-code capacity and reliability. Optimizing these parameters ensures efficient data encoding and robust error detection and correction capabilities.
6. **Checksums**: Incorporating check-sums into the bar-code data enhances error detection and correction during scanning. The choice of checksum algorithm and implementation should be optimized to balance data redundancy and bar-code efficiency.
7. **Testing and Validation:** Thorough testing and validation of the bar-code generation algorithm are essential to ensure that selected parameters and optimizations meet the requirements of specific applications. Testing should encompass a variety of scanning devices and environments to assess bar-code readability and reliability comprehensively.

3 IMPLEMENTATION WALK-THROUGH

3.1 PROGRAMMING ENVIRONMENT AND TOOLS USED

In the programming environment, Visual Studio Code (VSCode) served as the primary tool for developing the PDF417 bar-code generation code. Leveraging the capabilities of VSCode, I seamlessly integrated the ZXing.PDF417 package into the C project. However, the essence of the implementation lay in ensuring the incorporation of my equation, into the code base. This involved several adjustments and customization within the provided framework, aligning the generated bar-codes with the theoretical union equation.

3.2 DETAILED CODE WALK-THROUGH

The proceeding is a detailed explanation of the C# code that is used to generate PDF417 bar-codes.

3.2.1 Input Data

In order to pragmatically use the prior defined equation method this research is proving, first the input data must be defined an example will be used in this instance It begins by defining the input data, including the name, social security number (SSN), and Department of Defense (DOD) ID number. These data elements are encoded into the PDF417 bar-code.

```
string name = " John Doe"; string ssn = " 123
-45 -6789 "; string dod Id = " 1234567890 ";
```

3.2.2 Conversion to Binary

Next, convert the input data to binary format. This is achieved by iterating over each character in the input strings and converting them to their binary representations.

```
string binary Name = String To Binary ( name ); string binary Ssn
= String To Binary ( ssn );
string binary Dod Id = Convert. To String ( Convert. To Int32 ( dod Id ), 2).
        Pad Left (40 , '0 ');
```

3.2.3 Compression

After converting the data to binary, it is compressed to reduce the size before encoding it into the bar-code.

```
byte [] compressed Data = CompressString ( combined Data );
```

3.2.4 PDF417 Barcode Generation

Finally, the PDF417 is generated using the compressed binary data. The data is divided into rows, and each row is encoded separately to form the complete bar-code.

```
Bitmap bar-code = Generate PDF 417 Barcode ( compressed Data );
```

3.2.5 Concatenation of Row Images

In this step, the images of each row are concatenated form the complete bar-code image.

```
static Bitmap Concatenate Images ( Bitmap image1 , Bitmap image2 )
{
        Bitmap combined Image = new Bitmap ( image1 . Width , image1 . Height
                + image2 . Height);
        using ( Graphics g = Graphics. From Image ( combined Image ))
        {
                g. Draw Image ( image1 , 0 , 0);
                g. Draw Image ( image2 , 0 , image1 . Height);
```

```
        }
        return combined Image ;
}
```

3.3 TESTING AND VALIDATION PROCEDURES

In the testing and validation phase, the code was executed, debugged, and the resulting PDF417 bar-codes were generated and printed out for examination. This process ensured that the code executed as intended and produced the expected output.

3.3.1 Presentation of Generated PDF417 Bar-codes

Here is the generated PDF417 bar-code:

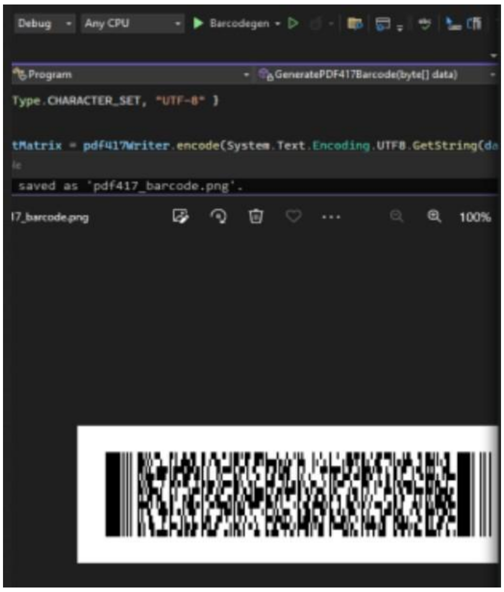

Fig.4. *PDF417 Bar-code*

4 RESULTS AND ANALYSIS

4.1 COMPARISON WITH EXISTING BARCODE GENERATION METHODS

The equation presented in this study for generating PDF417 bar-codes offers a novel approach that distinguishes itself from existing bar-code generative methods. While traditional methods often rely on predefined algorithms and mathematical mod- els, the equation proposed here offers a more flexible and adaptable framework. By emphasizing the dynamic composition of the bar-code through the union of rows, this enables greater customization and scalability in bar-code generation. Moreover, the equation’s incorporation of mathematical principles tailored to the specific requirements of Department of Defense (DOD) Common Access Cards (CAC) underscores its relevance and effectiveness in the realm of security technology. This comparative analysis highlights the innovative potential of the proposed equation in advancing the field of bar-code generation and enhancing security measures for sensitive applications.

4.1.1 Limitations and Challenges of the Proposed Algorithm

While the proposed algorithm for generating PDF417 bar-codes offers significant advantages in terms of flexibility and adaptability, it is not without its limitations and challenges. One notable limitation is the requirement to follow virtually the same steps for each cluster, albeit with a streamlined approach to enhance readability. While this simplification improves the algorithm's usability, it may also introduce constraints in certain scenarios where intricate customization is necessary. Additionally, great lengths were taken during the algorithm's development to ensure that it consistently outputs a valid bar-code. However, despite these efforts, the complexity of bar-code generation poses inherent challenges, including the need to balance readability, data capacity, and error correction. Addressing these challenges necessitates a delicate balance between algorithmic efficiency and bar-code quality, highlighting the ongoing need for refinement and optimization in bar-code generation techniques.

4.2 Suggestions for Future Research and Improvements

As the field of bar-code generation continues to evolve, there are several promising directions for future research and improvements:

1. One avenue for exploration is the enhancement of error correction mechanisms within the algorithm to improve bar-code reliability, particularly in environments prone to data corruption or interference.
2. Additionally, further investigation into optimizing the algorithm's performance could lead to advancements in bar-code generation efficiency, enabling rapid production of complex bar-codes with minimal computational resources.
3. Moreover, exploring alternative encoding schemes or incorporating emerging technologies, such as machine learning algorithms, could offer novel approaches to bar-code generation with improved data density and security features.

Collaborative efforts between researchers and industry stakeholders could facilitate the development of standardized benchmarks and evaluation metrics for assessing the effectiveness and robustness of bar-code generation algorithms, fostering innovation within the field.

DECLARATIONS

The following are the declarations presented by the author on the availability of the material as well as the funding"

5 CONCLUSION

5.1 IMPLICATIONS FOR BAR-CODE SECURITY AND TECHNOLOGY

The development of the proposed algorithm carries significant implications for bar-code security and technology. By providing a more dynamic and adaptable framework for generating PDF417 bar-codes, the algorithm enhances the resilience of bar-code-based security measures, particularly in applications requiring stringent data protection and authentication protocols, such as government-issued identification cards and product authentication labels. Furthermore, this emphasis on error correction and readability optimization contributes to the overall robustness and reliability of bar-code systems, reducing the risk of unauthorized access or data tampering. As bar-code technology continues to play an increasingly integral role in various industries, including logistics, retail, and healthcare, the adoption of advanced bar-code generation techniques offers opportunities for enhancing operational efficiency, supply chain visibility, and consumer trust. By leveraging the capabilities of the proposed algorithm, organizations can increase their security posture and unlock new possibilities for leveraging bar-code technology in a rapidly evolving digital landscape.

5.2 CLOSING REMARKS

Through the unveiling of the mathematical intricacies behind PDF417 bar-code generation, this study illuminates the symbiotic fusion of mathematics and modern security technology. As we navigate the evolving landscape of digital security, the insights gleaned from this research serve as a catalyst for future advancements in bar-code technology. With an unwavering commitment to exploration and collaborative endeavor, we stand poised to unlock new possibilities and address emerging challenges in the realms of digital security and information management, and journey towards a safer and more resilient digital landscape.

REFERENCES

1. Barcode Encoding - Introduction to ID card software - Learning Center — AlphaCard. Retrieved from https://www.alphacard.com/learning-center/intro-to-id-card-software/encoding-options/barcode-encoding/

2. CHIPS articles: Safeguarding your common access cards and military identification cards. Retrieved from https://www.doncio.navy.mil/Chips/ArticleDetails.aspx?ID=8042

3. DENSO WAVE. (n.d.). What is a 2D code? Retrieved from https://www.denso-wave.com/en/adcd/fundamental/2dcode/2dcode/index.html

4. Department of the Navy Chief Information Officer. (n.d.). CHIPS articles: Safeguarding your common access cards and military identification cards. Retrieved from https://www.doncio.navy.mil/Chips/ArticleDetails.aspx?ID=8042

5. FasterCapital. (n.d.). The science behind bar-code decoding algorithms. Retrieved from https://fastercapital.com/topics/the-science-behind-barcode-decoding-algorithms.html

6. Morovia Corporation. Overview - PDF417 Fonts & Encoder 5 User Manual. Retrieved from https://www.morovia.com/manuals/pdf417-font-encoder/chapter.overview.php

7. Updated plan for the removal of Social Security Numbers (SSNs) from Department of Defense (DOD) identification (ID) cards. Privacy, Civil Liberties, and Freedom of Information Directorate U.S. Department of Defense. Retrieved from https://dpcld.defense.gov/Portals/

8. VB.NET PDF417 Generator — generate, draw PDF417 bar-code Image in VB.NET applications with valid data input. (n.d.). Retrieved from https://www.onbarcode.com/vbnet/data-encoding/pdf417.html