AN ENGAGING GAME DESIGNED FOR THE EDUCATION PURPOSE OF K-12 BY INTEGRATING MATHEMATICS INTO GAME MECHANISMS

Xiang Li[1], Ang Li [2]

[1] Northwood High School, 7545 Portola Pkwy, Irvine, CA 92618
[2] Computer Science Department, California State University, Long Beach, CA 90840

ABSTRACT

Math Knight is a novel educational game designed to engage K-12 students by embedding mathematical principles directly into gameplay. This research investigates how integrating math into game mechanics—using a roguelike framework—can overcome common challenges such as math anxiety and disinterest. Developed in C# with the Unity engine, the game incorporates a dynamic health system represented by arithmetic expressions, knight ability to cast arithmetic operations, and adaptive difficulty tailored to individual performance. Experimental evaluations measure both shifts in students' attitudes towards math and improvements in mathematical, or more specifically, arithmetic proficiency. Preliminary results indicate increased motivation and arithmetic foundation. A comparative analysis with existing educational methods reveals that Math Knight not only supplements traditional learning but also aligns with the ongoing learning principle by fostering continuous conceptual growth. Although challenges such as visual design remain, the findings suggest that game-based learning offers a promising supplementary tool to enrich math education and stimulate a lifelong interest in the subject.

KEYWORDS

Game Design, Education Purpose, Integrating Mathematics into Game Mechanisms, Electronic Learning

1. INTRODUCTION

"Some of the heaviest criticisms of mathematics teaching and learning is the reliance on drill-and-practice as a pedagogy [18]."

"Mathematics is an important subject that is pervasive across many disciplines. It is also a subject that has proven to be challenging to both teach and learn…The use of games in a classroom can create a more exciting and engaging environment, while still reinforcing learning concepts [19]."
"The way in which students are being educated is evolving…There is a need to integrate more efficient mechanisms, like serious video games, in a non-formal environment for students and increase learning opportunities beyond the classroom [16]".

All the above quotes Indicate the importance of our subject: education with games. Experiencing the differences between Chinese education and United States education, I felt the crucial part to

David C. Wyld et al. (Eds): COMIT, AISO, CRBL, WIMNET, EDUPT– 2025
pp. 109-121, 2025. CS & IT - CSCP 2025 DOI: 10.5121/csit.2025.151209

set up a great mathematician is interest and practice. Chinese education lacks maintaining interest, and the US lacks practice [2]. I'm incredibly lucky to have built a strong mathematics foundation through practice while increasing my love for mathematics. However, not many people have the fortune to travel across countries and experience diverse education systems. As a result, I've wanted to help other young mathematicians to practice math and stimulate their interest. To achieve this goal, I've conducted studies and learned how virtual environments are just as effective as traditional hands-on learning [5]. With further evidence such as qualitative design showing that Educational Gaming Applications are effective in embedding science content into video games [6]. Then an idea struck my mind. I could combine mathematical principles and applications to a game format by employing my unique understanding of math.

As of late, I have begun to develop a game that combines the practice and logical thinking of mathematics into the game mechanics itself. My vision differs from current educational games in the market developed by educators, like Prodigy, which presents a single math problem that pauses the game and gives the students an unpleasant feeling about [21]. Specifically, I used my own understanding of mathematics to develop the game "Math Knight", which integrates math into the game mechanics, making the flow natural and making math enjoyable [3]. Not only will this game be entertaining to play, its core concept hides the feeling that the player is doing math, so they are learning without knowing it. This is a key concept that I want to have in my game, because if the players think they are learning, they may develop a mental block before playing. Game-based learning is an interesting solution, because it could serve as a supplementary tool and trigger interest in the subject [15].

In my experiments, I first tested how my game affected the players' opinion on math in my first experiment. The significant result was the number of responses that answered "useful" significantly increased. This was probably due to how my game mechanics involved the use of mathematics, which made them feel the use of arithmetic calculations handy.

In the second experiment, I tested out how my game helped with the players' understanding of mathematics, based on their performances of simple arithmetics. The results were optimal, most people(80%) scored high scores, 60% took less time to finish, and 47% finished faster with a higher accuracy. These results were fascinating to me, as I didn't know my game could have such a huge impact. I believe this happened due to the lack of practice prior to playing Math Knight. As the practice versus gain are really high at first, the students' practice in my game played a significant role to improve their arithmetic calculation skills.

My experiment aligned with past papers, showing how the effectiveness of intelligent tutoring systems(ITS) compared to that of regular classrooms showed that overall, ITS had no negative but small positive effects on K-12 students' mathematical learning [14].

2. CHALLENGES

In order to build the project, a few challenges have been identified as follows.

2.1. Computer Inaccuracy

When the player divides 1 by 3 for example, if the number turns into 0.3333, computers cannot store infinite digits, which means that it needs to round. But rounding creates problems, 0.3333 * 3 = 0.9999 it does not go back to 1, which is inconsistent with mathematical concepts. To overcome this challenge caused by computer inaccuracy, I would develop an algorithm that doesn't do direct calculations on numbers, but mimic how humans would simplify and calculate

fractions. This would make it easier for players to understand, and also introduce the mechanic of improper fractions.

2.2. Overcoming the Perception

One major component of my program is making math-based mechanics engaging for students who aren't naturally drawn to math. A key challenge is overcoming the perception that math is boring or difficult. To address this, I could use the storyboard and artwork of games to create an immersive world where math is integrated into the narrative, making it feel like a natural part of the adventure [7]. The content and plot could introduce compelling factors for the game.

Additionally, applying a roguelike style—where failure leads to learning and progression—could encourage persistence and experimentation. It also creates a sense of accomplishment every time the player levels up or gains a new skill.

2.3. Randomness

Another challenge is that My game contains many possible rewards for the player, generated randomly to provide the excitement of gaining legendary awards. But to have this randomness, I need a big pool of rewards. To avoid the big time sink to code each individual reward, I could implement a base class that takes in parameters for each reward card, to diversify their effect. I could use procedure abstraction to save time in the development phase, and make it easy to create new rewards in the future. This would greatly cut down development time, and reinforce good coding habits for myself.

3. SOLUTION

The game is programmed in c# with the Unity game engine [8].

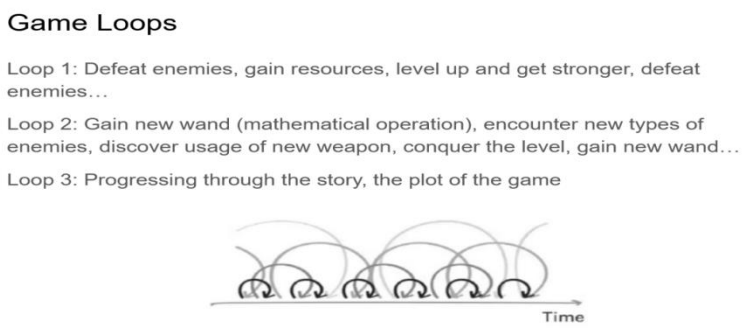

Figure 1. Game loops

One key game mechanic is that the player's health and the enemy's health are all represented using strings, and they won't die unless their health reaches 0. This creates a lot of space for creativity, which allows negative health and fraction health. Then after the player figures out how to manipulate the enemy's health to reach zero, the reward screen would pop up and the player will interact with the reward system to get stronger and progress through the game.

Level 1, player has the subtraction sword, enemies only have health with natural numbers$\mathbb{N}$={1,2,5,29…} [9]. This level prompts the player to familiarize with the game and practice basic counting with natural numbers. The boss has the ability to use addition and heal himself, which introduces the addition operation.

Level 2, player gains the addition axe, enemies health are now integers$\mathbb{Z}$={...,-25, -1, 6,10,...} This level introduces the idea of negative numbers to the player, allowing the player to master subtraction and addition with positive and negative integers. The boss has the ability to multiply the little enemies, which introduces the multiplication operation.

Level 3, Player gains the multiplication mace, enemies health are now rational numbers $\mathbb{Q}$={ …, -5, -½, ⅔ , 229, … } This level introduces the idea of fractions, prompts the player to use the multiplication operation to simplify fractions into integers. However, at the same time the player needs to think about the order of operations. For example, to defeat the enemy(69/5), if the player just multiply 5 without deep consideration, the enemy health would resolve to 69, which is hard to defeat; instead, the player will learn to perform minus operation first, a minus 13 would make the fraction ⅘, then multiply 5 and the enemy health would resolve to 4. By considering the order of operations, the player reduced the required minus from 69 to 17. The boss has the ability to divide the health of the player, which introduces the division operation.

Level 4, Player gains the division dagger. Enemy health still rational numbers $\mathbb{Q}$={ …, -5, -½, ⅔ , 229, … } However, enemy health increases significantly in number. This prompts the player to use the division operation because even 999 can be defeated within 6 operations, divide 3, divide 3, divide3, minus 2, divide 5, minus 7. Enemy health will also involve more advanced fractions with multiple prime factors, such as 1/90, which can be simplified to 1 by multiply 2, multiply 3, multiply 3, and multiply 5. This scratches on the concept of prime factorization. The player will gain significant practice of basic arithmetic operations and calculations finishing all four levels [12].

This adaptive learning creates a personalized study journey according to different individuals, which could have a significant impact on one's learning experience and gains [1].

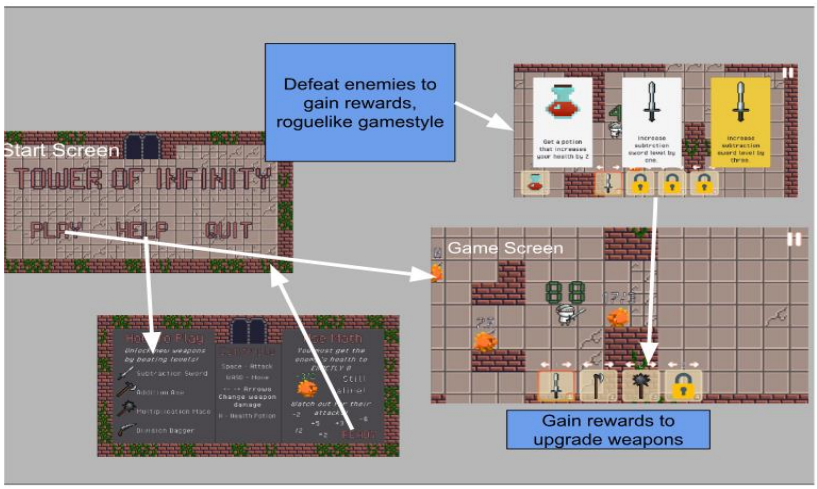

Figure 2. Overview of the solution

An integral part of my game is the way the player and enemy health is calculated. Instead of them being simple health systems, the player must get the enemy's health to exactly zero. This means that health can be negative. Not only that, but the player also has weapons to divide and multiply the enemy's health. This brings up strange situations; if any enemy has 6 health, and you divide it by 4, the health should be "6/4", instead of rounding down to 1. The way I accomplished this was by implementing an Expression Tree, which is a Binary Tree but instead of parent nodes representing more data, they represent the operation to do between its two leaf nodes [13]. This is a very unique concept, and very essential to my game.

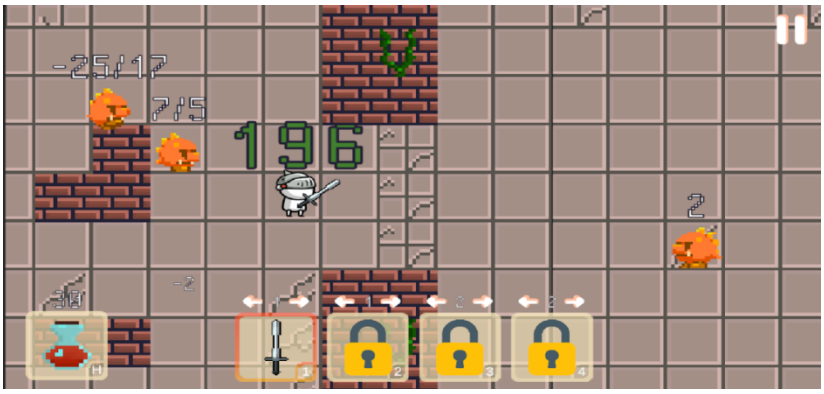

Figure 3. Screenshot of the game 1

```
private List<string> TokenizeExpression(string expression)
{
    List<string> tokens = new List<string>();
    string currentNumber = "";

    for (int i = 0; i < expression.Length; i++)
    {
        char c = expression[i];
        if (char.IsDigit(c) || (c == '.' && currentNumber.Length > 0))
        {
            currentNumber += c;
        }
        else if (c == '-' && (i == 0 || IsOperator(expression[i-1].ToString()) || expression[i-1] == '('))
        {
            // Handling negative numbers
            currentNumber += c;
        }
        else
        {
            if (currentNumber != "")
            {
                tokens.Add(currentNumber);
                currentNumber = "";
            }
            if (!char.IsWhiteSpace(c))
            {
                tokens.Add(c.ToString());
            }
        }
    }
    if (currentNumber != "")
        tokens.Add(currentNumber);

    return tokens;
}
```

```
private Node BuildFromPostfix(List<string> postfix)
{
    if (postfix.Count == 0) return null;

    Stack<Node> stack = new Stack<Node>();

    foreach (string token in postfix)
    {
        Node node = new Node(token);

        if (IsOperator(token))
        {
            node.Right = stack.Pop();
            node.Left = stack.Pop();
            stack.Push(node);
        }
        else
        {
            stack.Push(node);
        }
    }

    return stack.Pop();
}
```

Figure 4. Screenshot of code 1

The upper image code is the function used to tokenize a given expression. This gets called when the player/enemy deals damage, and breaks down the equation into a tree. For example, if the player has 6 health, and the enemy does a divide by 4 attack (6 / 4), the expression will break down into 3 tokens: 6, '/', and 4. These tokens will make it so the algorithm can simplify them as much as possible.

The lower image code shows the function that I used to build the actual ExpressionTree given a list of the expressions. After it is built, we can evaluate the tree to determine the simplest way to show its health. Since we don't want decimals, it breaks fractions down into their simplest forms before displaying them as health, keeping them as whole numbered fractions. This is crucial to my game to prevent decimal rounding errors.

The next component in my project is the Reward system for the player. Every time they defeat an enemy, they are presented with a random choice of rewards based on the condition of the game, and random odds. There are 3 different tiers of rewards, and each tier has a list of possible rewards, which all have unique and different effects at different strengths. Before generating a reward, I also take into account important factors like the players current health, potion value, and level. This makes the rewards more interesting and dynamic. It also adds a lot of replayability, because each level the player goes through will be different from the last, and each time they play the game on top of that.

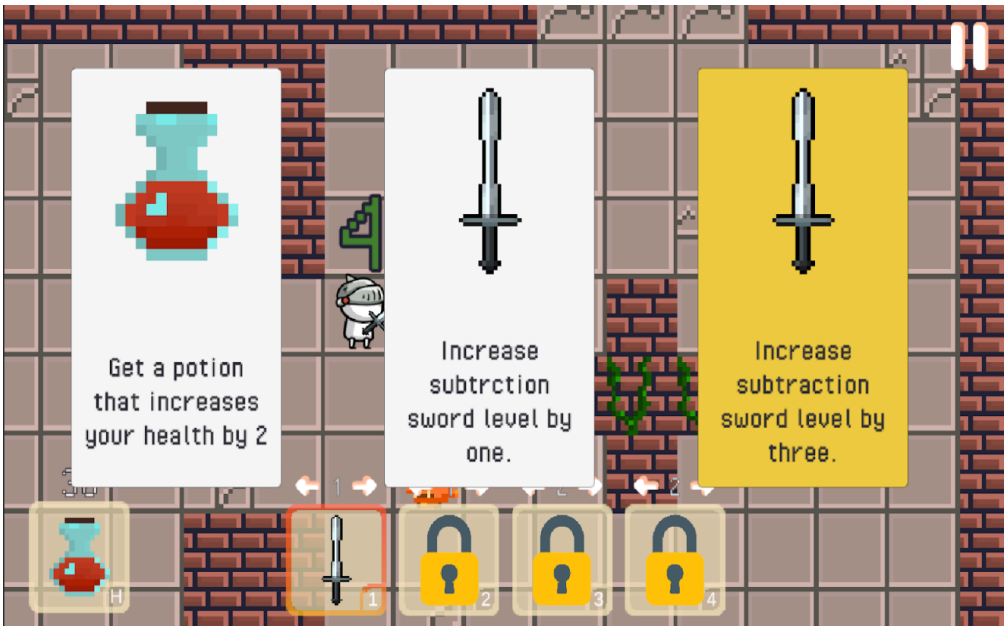

Figure 5. Screenshot of the game 2

```
1 reference
private GameObject GenerateRewardCard()
{
    int random = Random.Range(0, tier3Wieght+tier2Wieght+tier1Wieght);
    print(tier3Wieght+tier2Wieght+tier1Wieght);
    print(random);
    if(random< tier3Wieght)
    {
        return tier3Rewards[Random.Range(0, tier3Rewards.Count())];
    }
    else if(random< tier3Wieght +tier2Wieght)
    {
        return tier2Rewards[Random.Range(0, tier2Rewards.Count())];
    }
    else //tier 1
    {
        return tier1Rewards[Random.Range(0, tier1Rewards.Count())];
    }
}
0 references
public void OnRewardSelected()
{
    AudioManager.Instance.PlayOneShot(confirmSFX, 1f, UIamg);
    Reward();
}

6 references
public abstract void Reward();
0 references
public void Start()
{
    descriptionText.text = Description;
    buttonIcon.sprite = icon;
}
```

Figure 6. Screenshot of code 2

The above code is how I generate a random reward card. This is done three times every time the reward screen appears. First, it randomly chooses one of the three tiers to give, then from that tier, it picks a random possible reward. Also in the lower code, I used a base class that all rewards can inherit from. This made it much easier and cleaner for me to create lots of rewards by reducing the amount of code duplication I did. This is an extremely important part of my game, because it makes it more fun, and each time the player plays it results in a different experience.

Analyze player behaviour and generate enemies with health according to the players weakness. Provides more practice on the subjects that needs to be practiced on. This feature makes the learning opportunity more efficient. This component is not implemented yet, but I plan on doing it in the future. Below will be how to implement it in the future.

This component will be a great addition to the game, because it will help the player be constantly challenged, which is the key to learning. It would also be technically hidden to the player, which would make them learn without knowing they are learning.

Figure 7. Screenshot of lose page

When implemented, it will look something like this.

```
///how to change enemy health based on hits
///for each enemy store this
///    [ (timeHit, type), (timeHit, type), ...]
///Then from there, we have some important data:
///         # of hits it took to defeat a certain enemy
///              (this would be the amount of tuples in our list)
//               We can compare this to the optimal # of hits
///         #average time between hits.
//              A higher time may mean the player is slow at this particular type, and needs practice

//take all of these factors into consideration
//   create more enemeis with the type of math they are the worst at
//   For example, if the player's # of hits is consistently far above the average for multiplication type enemies, spawn more of them
```

Figure 8. Screenshot of code 3

The above pseudocode is roughly how I would store the needed information, and apply it to the game to make it more effective [11]. I will take into account the number of hits the player is using, and the enemy time which it's against, to spawn more enemies of the type of math the player is worst at. When implemented, this will help the player improve their skills on a certain topic without them even knowing it.

4. EXPERIMENT

4.1. Experiment 1

Now having the product, I want to test out if it actually works. The first part to investigate is whether the game was actually enjoyable and raises good opinions on the study of mathematics.
I conducted the experiment by hosting a summer camp, inviting 123 participants who are currently enrolled in elementary school in Irvine. This is the distribution of their specific grades in elementary school: 10 were 1st graders, 24 were 2nd graders, 27 were 3rd graders, 29 were 4th graders, 19 were 5th graders, 14 were sixth graders. A way we can collect data would be asking them which word best describes their feeling about math. Ask before and after, then analyze it like below:

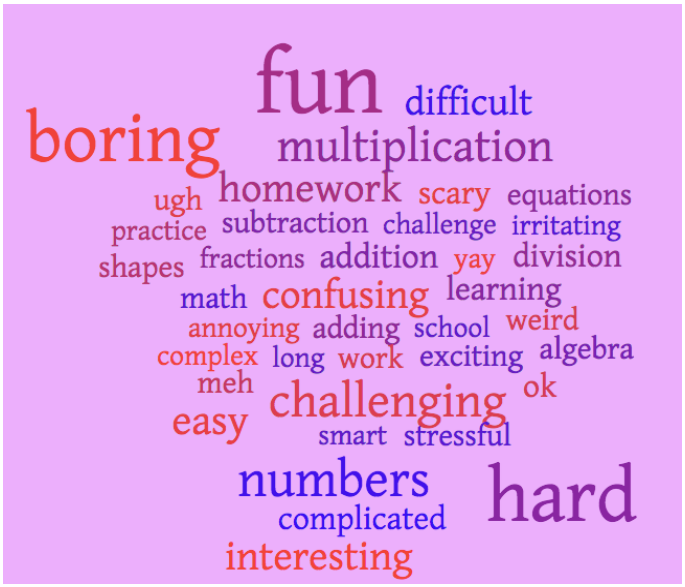

Figure 9. First set of data

This is a good way to visualize data, and makes it simple for me to know what players are thinking about my game, and how their perception of math changes throughout playing it.

Here is the first set of data collected when asking people for what comes to mind when they hear the word “Math”. Notice some of the most common words are boring, hard, and fun.

According to the data, when kids are asked about math prior to playing my game, their responses are mostly negative. With key words like “boring”, “hard”, “challenging”, and “confusing” showing up, it’s clear that math is seen as a negative thing. This is what I would have expected, it is no secret that math is seen in a negative way for most people at school. The feeling stems because kids are not shown why math is important, and the way they are taught mostly confronts the above words. The biggest effect on the results are the schools that are teaching math. It is not seen as interesting or fun, but more as boring and challenging. The next step to this experiment is to ask the players what their thoughts on math are after playing the game for a week in our summer program.

Figure 10. Data after one week of Math Knight

After experience with our game, negative voices still exist, but the majority of the words that the kids used to describe their feelings are: “rewarding”, “awsome”, and “learn”.

This piece of data is strong evidence of how my game Math Knight changed the perspectives students had on math. The words show a clear contrast with the word cloud in Figure 9, indicating a shift of tone from negative to positive. This result shows how engaging learning by integrating math into game mechanics instead of inserting math like advertisements can induce the game-like interesting nature of math, generating an exciting learning experience for kids in elementary schools.

4.2. Experiment 2

The second experiment I want to test out is whether the game actually helps with the practice of mathematics. It is crucial that this game is not just for fun, but has a real impact on math skills.

I invited (some number of) students to participate in my organized summer camp. I first let them take a quick test on their current knowledge of simple arithmetic calculations. I record the time

and accuracy for each test as experimental data. I then compared their prior data to their data after playing the game to determine if it is making progress towards the goal of helping them learn math.

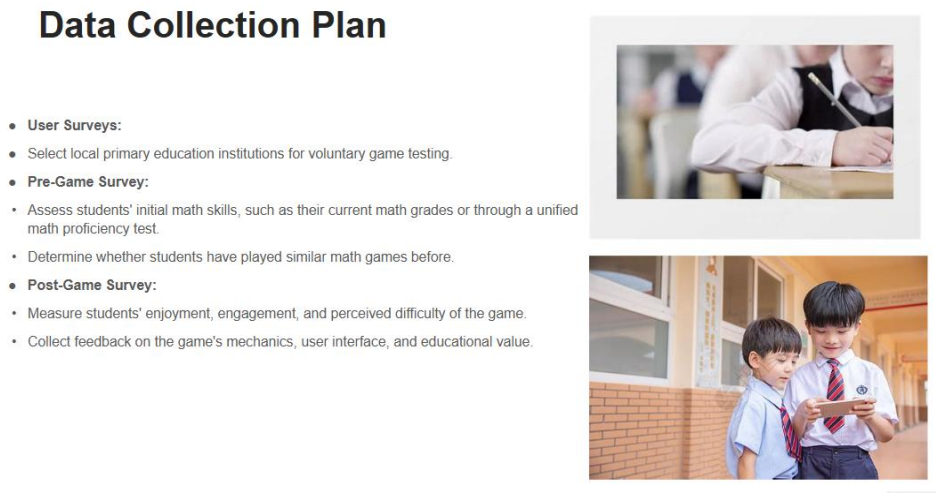

Figure 10. Data collection plan

5. RELATED WORK

5.1. Methodology 1

Method: Video Stimulated Recall

Fails to accommodate the "Ongoing Learning Principle". "Within the context of mathematics education, such a principle is useful in terms of expansive learning where the student will revise previous learnings and concepts and modify these in light of new learnings. A simple example of this is as the students develop proficiency with additional facts, they develop a schema for addition being an increasing process but as this proficiency becomes an automatic process, negative numbers are introduced and the increasing schema must be modified to accommodate the learning. The concept for addition is substantially revised to incorporate new experiences. Learning occurs on a particular trajectory where principles can be intuited and developed into more complex principles as further learning experiences are provided. The scaffolding to enable on-going learning supports the current learning but in subtle ways also expands learning to include anomalies so as to build new, more complex understandings [18]."

A sample 2012 video game teaching mathematics aimed to target the new generation of "digital natives" Motivation to play, but learning was limited because it was possible to do trial and error. My project improves by cleverly integrating the uses of mathematics in a roguelike game. The nature of leveling up and progressing through harder levels in roguelike games aligns perfectly with the "Ongoing learning Principle".

5.2. Methodology 2

They concluded that there are positive effects to gaming, and proposes some digital games that resulted in positive education effects.

1st game: Dinner for dogs [19]: role-playing interactive game where the player assumes the role of a dog caretaker.

tasked with feeding dogs a certain amount of food with limited resources
Boring due to the flow. The reward player gets for correctly feeding the dog is only the dog being happy. Not attractive and would be boring after a few plays.

My method avoids that with the reward system that uses randomness, provides excitement and expectations, with engaging feedback of level up and getting stronger.

2nd game: The last chip game [19]

A puzzle game with the objective to take the last chip under various rules. Similar to the NIM game, it aims to exercise mathematical reasoning [10].The game structure and objective limits the thing that a student can learn from it. NIM games aren't a major subject in school. And again, lacks motivation due to the lacking reward system.

My method avoids all these, with fundamental arithmetic operations in the field of Rational numbers that would clearly benefit students in school. And again, with an addictive roguelike game style.

5.3. Methodology 3

The third methodology conducted on the role of computer games in education uses Three different types of research to analyze how computer games in education can develop [17].
Value-added research suggests five promising features to include in educational computer games: modality, personalization, pretraining, coaching, and self-explanation.

Cognitive consequences research suggests two promising approaches to cognitive training with computer games: using first-person shooter games to train perceptual attention skills and using spatial puzzle games to train two-dimensional mental rotation skills.

Media comparison research suggests three promising areas where games may be more effective than conventional media: science, mathematics, and second-language learning. Future research is needed to pinpoint the cognitive, motivational, affective, and social processes that underlie learning with educational computer games.

However, this article was published in 2019, and by that time, computers weren't as prevalent in education as nowadays. Furthermore, it analyzed potential methods to provide valuable games for education, but it didn’t create any product or recognize any games that meet the above criterias. Its limit is evident in “ Future research is needed to pinpoint the cognitive, motivational, affective, and social processes that underlie learning with educational computer games [17]”. All the 3 methodologies above address math education in the gaming industry. They try to accomplish an optimal outcome of integrating games with math. However, the attempts were either too boring on the game side, or too trivial on the math education side. The third methodology provides guidance on how to create a good math game, but didn’t provide a product that meets its criteria.Thefister tried video stimulated recall, but was too boring for kids. The second approached in three different ways, but some were trivial math concepts that wouldn’t benefit the kids much in their everyday study.

My project did the most basic Four arithmetic operations which will always be useful in math, providing practice on core math concepts, while integrating the use of mathematics into the game

mechanics, combined with a roguelike style, which makes the game itself attractive. Taking away the math part, it would still be a fun game.

6. CONCLUSIONS

The strength of my game is the high integration of mathematics with the game mechanics, which provides the players with a virtual learning experience that reinforces knowledge [4]. The roguelike game play encourages repetition and practice [20], an important factor of solidifying knowledge.

Some limitations to my game Math Knight are the artwork of the game. The animations aren't ideal, but could be made more appealing to kids. Another limitation might be some require of patience, it is a turn based game, so the player can't bash through.

For the future plan, to encourage more strategic thinking and promote deeper learning, introduce an attack counter that tracks how often players use weapon operations, discouraging overuse through thoughtful gameplay. A competitive ranking system based on the number of attacks used to complete levels can further motivate players to improve. Additionally, implement adaptive AI that analyzes player performance to generate enemies that exploit their weaknesses—offering targeted practice without overtly signaling it, making learning seamless and engaging. Finally, expand the arsenal with new weapons based on exponential and radical operations, such as square, square root, and their upgraded forms, cube and cube root, adding both variety and mathematical depth to the gameplay.

In conclusion, the question I am addressing is how to integrate math education in the game form so that it could be compelling to all children and encourage practice. Math Knight is my solution to the question. My product serves as a supplement outside the classroom that could help kids learn and practice math in a more fun environment anytime they want.

REFERENCES

[1] Darrah, Marjorie, et al. "Are virtual labs as effective as hands-on labs for undergraduate physics? A comparative study at two major universities." Journal of science education and technology 23 (2014): 803-814.

[2] Tatli, Zeynep, and Alipasa Ayas. "Effect of a virtual chemistry laboratory on students' achievement." Journal of Educational Technology & Society 16.1 (2013): 159-170.

[3] Balamuralithara, Balakrishnan, and Peter Charles Woods. "Virtual laboratories in engineering education: The simulation lab and remote lab." Computer Applications in Engineering Education 17.1 (2009): 108-118.

[4] Yaron, David, et al. "The ChemCollective—virtual labs for introductory chemistry courses." Science 328.5978 (2010): 584-585.

[5] Darrah, Marjorie, et al. "Are virtual labs as effective as hands-on labs for undergraduate physics? A comparative study at two major universities." Journal of science education and technology 23 (2014): 803-814.

[6] Annetta, Leonard A., Meng-Tzu Cheng, and Shawn Holmes. "Assessing twenty-first century skills through a teacher created video game for high school biology students." Research in Science & Technological Education 28.2 (2010): 101-114.

[7] Potkonjak, Veljko, et al. "Virtual laboratories for education in science, technology, and engineering: A review." Computers & Education 95 (2016): 309-327.

[8] Deshpande, Amit A., and Samuel H. Huang. "Simulation games in engineering education: A state-of-the-art review." Computer applications in engineering education 19.3 (2011): 399-410.

[9] Kuenzi, Jeffrey J., Christine M. Matthews, and Bonnie F. Mangan. "Science, technology, engineering, and mathematics (STEM) education issues and legislative options." Progress in education (2006): 161-189.
[10] Bouton, Charles L. "Nim, a game with a complete mathematical theory." Annals of mathematics 3.1/4 (1901): 35-39.
[11] Oda, Yusuke, et al. "Learning to generate pseudo-code from source code using statistical machine translation." 2015 30th IEEE/ACM International Conference on Automated Software Engineering (ASE). IEEE, 2015.
[12] Merrikh-Bayat, Farnood, and Saeed Bagheri Shouraki. "Memristor-based circuits for performing basic arithmetic operations." Procedia Computer Science 3 (2011): 128-132.
[13] Wang, Lei, et al. "Translating a math word problem to an expression tree." arXiv preprint arXiv:1811.05632 (2018).
[14] Steenbergen-Hu, Saiying, and Harris Cooper. "A meta-analysis of the effectiveness of intelligent tutoring systems on K–12 students' mathematical learning." Journal of educational psychology 105.4 (2013): 970.
[15] Tobias, Sigmund, J. Dexter Fletcher, and Alexander P. Wind. "Game-based learning." Handbook of research on educational communications and technology (2014): 485-503.
[16] Francisco J. Álvarez-Rodríguez, Arturo Barajas-Saavedra, et al. "Serious Game Design Process, Study Case: Sixth Grade Math." Creative Education Vol.5 No.9 (2014).
[17] Richard E. Mayer. "Computer Games in Education" Annual Review of Psychology Volume 70, (2019): 531-549
[18] Robyn Jorgensen (Zevenbergen), Tom Lowrie. "Digital Games for Learning Mathematics: Possibilities and Limitations" Annual Meeting of the Mathematics Education Research Group of Australasia (MERGA) 35th, Singapore (2012)
[19] Gul Ayaz, Katherine Smith. "Digital Game-Based Approach to Math Learning for Students" Modeling, Simulation and Visualization Student Capstone Conference (2023).
[20] Szabados Gy. N, et al. "Roguelike Games-The way we play" International Journal of Engineering and Management Sciences (IJEMS) Vol. 7 No. 4 (2022).
[21] Brooks, Christina D. "Prodigy Game and Third-Grade Mathematics Achievement in an Urban Setting" Walden University ProQuest Dissertations & Theses (2022).