Learning While Producing: The Use of Games in Teaching Statistical Process Control
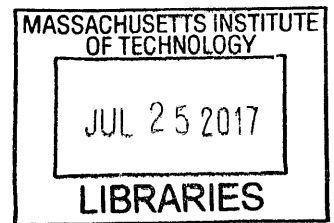
by

Jane S. Coffrin

Submitted to the
Department of Mechanical Engineering
in Partial Fulfillment of the Requirements for the Degree of

Bachelor of Science in Mechanical Engineering

at the

Massachusetts Institute of Technology

June 2017

Signature of
Author:

Signature redacted

Department of Mechanical Engineering
May 15, 2017

Signature redacted

Certified by: _____

David E. Hardt
Ralph E. and Eloise F. Cross Professor of Mechanical Engineering
Thesis Supervisor

Signature redacted

Accepted by: _____

Rohit Karnik
Associate Professor of Mechanical Engineering
Undergraduate Officer

Learning While Producing: The Use of Games in Teaching Statistical Process Control

By

Jane S. Coffrin

ABSTRACT

Statistical Process Control is used daily in factories throughout the world and its use improves the quality of manufacturing parts. But while it is widely used, it is not widely understood by the workers implementing these techniques. Without a strong understanding of Statistical Process Control, it can be misused and cause more harm than if it wasn't implemented. Correctly implemented Statistical Process Control, however, can improve the quality of the implemented parts. To remedy this, a game was designed to teach people with limited exposure to manufacturing and/or statistics an understanding for how Statistical Process Control is used in manufacturing. This game implements a mathematical model that takes factors within the game and raises or lowers the variance based on in-game factors then random gives a number based on the expected values. A paper prototype for this game was made and tested with students from multiple majors, then was turned into a python program. A pre-test and a post-test were developed and then used to test the base knowledge of the participants as well as their learning after playing the game. From the observations of participants playing the game and from their pre-test and post-test results a set of next steps was developed to make this a more viable game for teaching workers about Statistical Process Control.

Thesis Supervisor: David E. Hardt

Title: Ralph E. and Eloise F. Cross Professor of Mechanical Engineering

Acknowledgements

I would like to thank Professor David Hardt for advising my thesis and sitting down with me on a weekly basis to make it a success. As well as thank Professor Jose Pacheco for his assistance and presence in those weekly meetings. Without both of their insights into the problems in the manufacturing world this thesis would have never occurred. I'd like to thank all my friends who took the time out of their day to test this game. Extra thanks to my friend Suzanne Mueller, who play tested the paper prototype version of my game and was so excited about the project used her python skills to make it digital. I would also like to thank my academic advisor Dawn Wendell for the support she has given me throughout my time at MIT. Finally, I'd like to thank my family for their endless support and interest in my academics.

Table of Contents

List of Figures

1. Introduction

1.1 Motivation

This project looks to investigate one way to improve understanding of the principles behind statistical process control. Workers in factories use statistical process control on a daily basis, but many lack the formal education as to why they use statistical process control. This project looked to teach a complex technical concept in a nontraditional way that didn't require an understanding for complex math or high literacy. The end goal was to improve understanding about statistical process control regardless of background of the worker.

1.2 Teaching Advanced Concepts to those without Advanced Backgrounds

An important part to this project is making sure that it can be taught to anyone. This isn't easy to do, you gradually learn more math and reading as you go through school for a reason.

One example is from Antonio Ruiz-Martinez et al. from the University of Murcia where they researched and developed a laboratory to teach an advanced concept in computer networking to their students [1]. The laboratory they created is an interactive learning tool where students can follow along with a set of instructions and be able to replicate the processes with room to make mistakes, make their own improvement, and learn about this advanced computer networking concept. They had been using and

polling students about this laboratory and found that students enjoyed the activity and felt that they were learning more as well as a trend of increasing grades that aligned with the goal of improved learning.

Another example is from John Hunter, an elementary school teacher in Virginia, created a game called "The World Peace Game" [2]. Every year he changes the rules and scenarios to match the current state of the world and every year he plays this with his class of 4th grade students. His class is given a set number of days with the end goal of world peace. The students are given a list of problems that they need to solve in this time period to be successful, but in addition to solving these problems to be successful each faction must start with more money than it ends with. These 4th graders take turns making moves, debating, and discussing potential solutions. World peace isn't meant to be a simple solution, but this dynamic game shows that even complicated topics can be taught through games. In fact, it shows that games can facilitate learning in a way that the traditional classroom might fail. But more than show the potential use of games, it also displays to possible to teach children concepts that are out of the scope of what they should be able to. The World Peace game provided an opportunity for these students to learn in depth about complex topics that can be difficult to understand.

1.3 Introduction to Statistical Process Control

Statistical Process Control is a method of controlling the quality of manufactured parts by looking to achieve a truly random process by controlling or eliminating deterministic sources of error. Across the world Statistical Process Control or SPC is used to limit the error in manufacturing and create the best parts possible. Workers in

these factories are using SPC techniques on a daily basis, but not all of them understand the math and reasoning behind what they're doing [3]. This lack of understanding can cause big problems and cause SPC to be misused in factories, which can be costly for companies [4]. The incorrect use of SPC by companies can be further amplified by workers that lack training in SPC because it is typically only taught in the classroom as part of a formal course, from an online course, or a book. And while these are great opportunities to learn more about SPC they often aren't requirements to get a degree and many require a high level of math in order to participate, but not every worker in the factory may possess that level of math education. Expecting a worker to sit through a formal class or read a dense book and learn what they need to know is time consuming and potentially only effective if they have a math background. Ideally, however, every worker, regardless of background, would have an understanding about why they use SPC on a daily basis. Factory workers often have lots of down time while a work, so a custom solution for them could be developed to keep them engaged and learning in their down time for current factory workers. For future factory workers, this game should be used before the employee begins to work on the factory floor.

1.4 Current Methods for Teaching SPC

Statistical Process Control is often not taught to many unless they are taking a manufacturing course, but sometimes isn't even discussed in the introductory to manufacturing class. Outside of the university classroom setting there are few opportunities to learn more about SPC. Todd Abel and Lisa Poling created an activity for

high school age students to learn more about statistical process control that could be facilitated by a teacher [5]. The activity is called "Hold My Calls" and asks high school students to collect data, draw and interpret graphs, and discuss principles of statistics [5]. To learn more as a worker in the field, the options are fewer and drier. Companies can hire another company such as Blue Agility to give a one to four day in-person training on SPC and its uses [6]. They can independently take an online course, with a variety of paid and free options available such as a section of the Operations Management class for the online learning resource Coursera [7] or the paid class offerings from ASQ [8]. And finally, they can grab a book on statistical process control and read that on their own time. Unless workers were graced with a high school that valued the concepts of probability and statistics, they likely missed out an opportunity to learn about SPC in a non-traditional way. For this thesis, the using games to teach SPC was tested as a non-traditional method to educate existing workers during down time and future workers before they start working.

1.5 Introduction to Educational Games

People have been playing games for centuries, but when video games were first developed they were not seen favorably. Video games were initial brushed aside because too much screen time would ruin your brain and video games had a reputation for being too violent. But as time has passed, more people have warmed up to games. In fact, many people play video games, though not all would admit to being gamers. A great example, is how the Facebook game FarmVille captured all ages and had them checking their crops multiple times a day to be successful[9]. As these barriers between society

and video games break down the educational possibilities of games have begun apparent. Games, both digital and non-digital, have the potential to be educational tools and provide players with learning opportunities. Games can enhance the classroom experience by reinforcing taught material or they can teach something new that is often overlooked in the classroom setting. The interactive opportunity that these games provide offers a different take on learning that might otherwise be ignored.

The biggest danger with educational games or "serious" games is that they can be used as simply tools to continue to teach the same ideas. For example, a game that teaches addition, but the same teaching style is used. The only change is the interface, you're counting cookies or another fun object, this visual and interesting object is supposed to suddenly make addition fun. But it does, this is what Professor Scot Osterweil, would call "chocolate covered broccoli" [10]. While the outside visuals and objects may be appealing, it really doesn't make addition any more fun or easy to learn. To overcome the danger of creating a "chocolate covered broccoli" game, there must be the opportunity to make choices [10, 11]. Choice makes learning happen, it gives the player the opportunity to learn from their mistakes as well as forge their own path to success [11]. Choice makes the game like a choose your own adventure book, there is more than one path to the finish.

2. Game Design

2.1 Critical Game Requirements

For educating workers on Statistical Process Control to be a success, key takeaways were developed to focus on the most essential learning objectives that need to be incorporated in a solution. An ideal solution would allow people who use SPC daily with a limited math background to understand why they're using it and how to correctly implement it. This is important because the goal is to teach the principles of SPC to anyone in as convenient fashion as possible. This also means that a new university class would not work because it would pull workers away from the factory. An online class would also likely be ineffective for these purposes as it would likely require a formal math education to parse the material. And detailed written document would also be unacceptable as it would require a high level of reading comprehension. An ideal solution would allow the worker to interact with the program and use it during occasional downtime on the job. Creating a game was one way to satisfy this interactive condition that could be used during down time with little need to parse technical information. Games have been around for centuries, but with the development of video games, a new realm of gaming had begun. Now that time had passed, games were seen as a way to teach things to people. The key to making this game would be to focus on the interactivity of the module.

2.2 Statistical Process Control Concepts

To inform workers about Statistical Process Control, a few key concepts were selected as learning objectives. These are understanding variance, mean shift, as well as understanding how to eliminate possible sources of error. Variance is the variability and inconsistency from one part to another, a system with a higher variance will have a larger range of values that parts can fall into. The mean of the system is where the average value of parts fall and a mean shift is when the mean of the system changes, therefore the average value of parts shifts. Finally, eliminating error, while this may be done in a factory by improving processes, preventing machines from overheated, and regulating worker breaks, there are many other ways to control error. Any game that is developed should keep in mind these three learning objectives and look for a way to teach users about variance, mean shift, and eliminating error.

2.3 Game Concepts

Two initial game concepts were developed around concepts of statistical process control. The first used dart gun shots on a target to teach the concepts of variance and mean shift. Players would start with the most basic set of equipment, an old gun, a sight made from a toilet paper roll and would shoot in their own backyard. Players could earn ten dollars in game money for each shot that hit within the two inner rings of the target. With the money players obtained they would have the option of how to spend it, giving them the choice of choosing their preferred upgrade path. Players have the opportunity to buy a new dart gun, a new sight, or buy a membership to an upgraded shooting range. The end goal of this game was to show what a truly random process would look like once

sources of error had been controlled, in this case by buying upgrades that improved the shooting process. This game provided the opportunity to provide learning opportunities in all three SPC learning objectives.

The second concept was focused on cooking, in fact, the initial idea stemmed from trying to cook tomato soup. But this was no ordinary tomato soup, it was the player's Grandmother's famous soup. The problem? The recipe has gone missing and it is up to the user to recreate it. Players advance through rounds by trying different ingredients such as choosing between skim milk and whole milk and choosing ingredients such as the amount of salt to add. As players go through rounds experimenting with their recipe, they enter tomato soup competitions. As they successfully guess the correct ingredients they can advance to the next round of competition before have to appease all their relatives who have tasted their Grandmother's original tomato soup. Each judge would have three characteristics to judge, such as creaminess, texture, and seasoning. Because taste varies from person to person there would be some inherent variety in their taste and could even vary from batch to batch. This idea focused more on the learning objectives of eliminating error and variance, but less on mean shift. In the end, the concepts taught in the first proposed game better aligned with the desired learning objectives, so the dart gun shooting game was the choice moving forward.

2.4 Initial Game Design

When designing the game to reflect real life SPC, 3 controllable variables were selected: the shooting range location, the quality of dart gun, and the selected sight. These controllable variables would have an effect on the mean location and variance of

the shots by the player. The player would have the opportunity to explore different upgrades to see the improvement of their shooting quality with various upgrades. Not every upgrade impacts both variance and max mean shift. It was different for each type. Changing the sight decreased the variance of each shot, but changing the gun impacted both. Changing shooting ranges from backyard to outdoor range impacted only variance, but from outdoor to indoor range impacted only mean. This is meant to account for factors such as the weather or windy conditions. In addition, another choice was given to the player who would shoot 10 shots in a round. The player can choose the number of shots to shoot at a time until they reach the shot limit. Shots shot at the same time, have the same calculated mean value that each shot falls based on its randomly generated value, which is based on a normal distribution curve. If a player chooses to shoot five shots at once, then shoot another five, that is equivalent to shooting their dart gun five times, putting it down, picking it up again to shoot five more shots. This change in procedure impacts the location of the darts as they aren't expected to fall in the same range. Players shoot at a target as shown in Figure 1. and earn money by getting darts to land in the inner two rings. For every shot that lands in the inner two rings the player earns $10, which can be used towards paying for future upgrades.

Dartgun: superior 1, Location: indoor, Sight: laser

Points: $550

Shots Left: 0

10|

Shoot

basic: $0   midrange: $100   superior 1: $250   superior 2: $200

toilet paper: $0   red dot: $150   laser: $250

backyard: $0   outdoor: $50   indoor: $200

Next Game

Figure 1. A view of the user interface of the game, where a player has just taken 10 shot in succession. The darts are represented by the green dots. Players can see the items they have equipped just below the target and can buy other upgrades as they earn money by clicking on the buttons below. This is the target of a player that has bought all the upgrades.

2.5 Methodology

To actually place points on the target, a mathematical model was created that emphasized the true randomness of the system. First, based on the shooting range, gun, and sight a maximum value was calculated. This value was used as a calculated mean point for the shots. The term calculated mean was used because it doesn't reflect the true mean of the one to ten generated points, but would be an accurate mean for an infinite number of shots. It was placed away from the center at a randomly generated angle. Then based on the variance of the selected equipment the shots were given a distance from this calculated mean and an angle to be placed at. While most of the numbers in this system were completely randomly generated, the random number for variance was generated on a normal distribution curve, which allowed for the possibility for outliers. The mean is reselected for subsequent shots that aren't taken in a row to show the difference that just placing the gun down can have on shot location. Each gun, sight, and location was associated with a value for mean and standard deviation as shown in Figure 2. Based on the selected equipment the value, the mean and standard deviation was chosen by adding the values from the various upgrades together. To fit these numbers, which ranged from 3.2 to 0.85, on the target the added radius of each ring was 0.5, with the center being at 0. For more information on the mathematics behind making this game work, the python code is available in Appendix A.

	Name	Mean	Std. Dev.
Location	Backyard	1.20	1.25
	Outdoor	0.70	1.25
	Indoor	0.70	0.60
Gun	Basic	1.00	0.90
	Mid-range	0.50	0.65
	Superior1	0.20	0.35
	Superior2	0.30	0.25
Sight	Toliet Paper Roll	1.00	0.00
	Red Dot	0.50	0.00
	Laser Sight	0.10	0.00

Figure 2. A table showing the values of the mean and standard deviation associated with each possible upgrade. Each upgrade impacted mean and standard deviation differently.

2.6 Pre-Test and Post-Test Creation

To test the effectiveness of the game as a learning tool, a pre-test and a post-test were developed. The pre-tested asked general questions about the participant's education such as college and their coursework familiarity with manufacturing as well as statistics. After obtaining that baseline for knowledge, they were quizzed on a few short questions that were primarily multiple choice to determine how much they knew or remembered about SPC. One question on the pre-test even asked the participant to define SPC in a short sentence or two. The participant then played the game for a

maximum of 20 minutes or through 25 rounds of shooting. After testing, the participant was then asked to reflect on their in-game experience through a series of question that mirrored the first questions they were asked as well as some questions that were more specific to the game. The in-game questions asked how they thought different factors impacted the mean vs how they impacted variance and standard deviation. These pre-test and post-test questions can be found in Appendix B. The two results were then compared to determine if the learning and understanding of the participant had improved from playing the game. Ideally, this would be tested using this method with a wide range of backgrounds and education levels.

3 Play Testing

3.1 Pre-Test and Post-Test Results

The darts game was tested on twelve undergraduate MIT students from various majors. There were eight Mechanical Engineering majors tested and five from a variety of majors including Biology, Political Science, Business, and Computer Science and Electrical Engineering. However, due to the technical prowess of MIT it is likely that all of these participants have been introduced to basic statistics at some point before and are not the ideal candidates to test this game, a future study should look for participants with a less technical background. Of all those tested, four had taken a manufacturing class at MIT, three had taken a statistics class at MIT, but only one participant had taken both a manufacturing and a statistics class at MIT. Mechanical engineering majors did not do better than their peers who were science or humanities majors on the pre-test, but those who had taken a manufacturing or statistics class at MIT were more likely to get a high pre-test score, with four out of six of these students acing the pre-test. Of those tested, six received a perfect pre-test score and seven received a perfect post-test score, but only two participants received both a perfect pre-test and post-test score.

After playing the game and taking the post-test, not everyone's test scores improved or remained the same, in fact, four participants saw their score from the pre-test to the post-test decrease. Most of those who didn't have perfect scores from before, improved their scores after playing the game and answering more game specific questions, suggesting that playing the game actually helped these users learn and understand the statistics principles being asked. But the more noticeable (and perhaps

alarming) group was those who got perfect scores the first round, struggled when asked about in game questions and their scores slipped from their pre-test results, many of these people had taken MIT manufacturing and statistics classes and it raises the question that they might have a good mathematical understanding of mean and variance, but not a practical sense. Of course, with such small sample sizes, it is entirely possible that this is not the case, but notes an area worth investigating. Compiled results can be found in Appendix C.

3.2 Game Play Observations

While players interacted with the game, each settled on a shooting strategy and a purchasing strategy. Three shooting strategies developed during game play after a few rounds of experimentation with the interface. There were Slow and Steady shooters that shot one bullet at a time until their ten bullets for the round were up before repeating the same process again. Then there were the Rapid Fire shooters that shot all ten bullets at the same time every time, hoping to capitalize on good aim when they pick up the dart gun. Finally, was the Split the Middle shooters that shot three to five bullets at a time, in hopes to capitalize on good aim, but not waste all their shots if they happen to have poor aim for their first set of shots. The two purchasing strategies that developed were of a buy immediately and a save and invest. Some participants were inclined to buy the upgrades as soon as they had the money to purchase for a new gun, sight, or location and bought their upgrades in that fashion. The other most common strategy for purchasing upgrades was to hold out on buying any upgrades upfront and instead save money to buy the better upgrades once the money was available. Players who followed

the buy first strategy were generally able to purchase the most of the advanced upgrades quicker than those who tried to hold out on buying upgrades to get the more advanced ones. A complete set of 25 rounds for a Slow and Steady Shooter and a Rapid Fire shooter can be found in Appendices D and E.

Figure 3. A side by side comparison of a Slow and Steady shooter (left) and a Rapid Fire shooter (right). The different strategies show up differently on the target. A Slow and Steady shooter has more variance in where their shots land, but they have more shots in close proximity to the inner two rings. A Rapid Fire shooter has less variance in where they shoot, but that doesn't means their shots are near the center.

3.3 Key Takeaways

While this game appeared avoid the desired pitfall of becoming chocolate covered broccoli, there is still a long way before it can be used as a useful tool for educating workers on statistical process control. The game was effective in offering the

participants the opportunity to make choices to develop their own strategy and provided multiple paths to victory, which allows for the player to feel satisfaction. But it lacked features that could have enhanced the educational portion of game as well as features that could have made the game more engaging. Both the educational and interactive enhancements should be considered in future iterations of the game and will be described in more detail in section 4.

4 Game Iteration

4.1 Immediate Changes

To improve the game, it should be made both more educational and more fun, as well as made easier to understand. Players requested a round counter, so that they could digitally keep track of the number of rounds they had played through instead of manually. Players also requested a clear way to win either by winning a competition or buying all the upgrades, but something clear that offers the satisfaction that they had succeed, rather than just playing until twenty-five rounds had passed. One potential solution to this is to create the opportunity to go to competitions and compete against two computer players, if the player shoot the best of the three then they can get a money bonus and a medal. They could work up to a final competition at the National Dart gun World Championship to win the overall game. Alternatively, a simpler potential win state is to have all ten darts hit the inner two rings of the target, this could be more simple implemented from the current code, but has a chance, albeit small, to cause the player to win early on if they get unusually lucky with their shots. To address adding educational benefit, players should be provided with more information about the mean and standard deviation (and perhaps variance) of their shots. This can be implemented with a line or two of text below the target stating the location of the center of the shots and the standard deviation of the shots. Alternatively, and possibly more effective for the target audience it can be shown graphically on the target with a different colored dot representing the location of the mean and with a circle depicting the location of one standard deviation from the mean value.

4.2 Future Changes

It's difficult to suggest other future changes to improve the educational value of this game Changes to this aspect should be done based off future test player surveys to check for their understanding of statistical process control. But one change that could be implemented to make the game feel more game like is the ability to have some control over accuracy. This could be implemented with a bar, like shown in Figure 3, that has different colored zones with a triangle sliding up and down it. The user would then have to hit space when they wanted the triangle to stop moving based on the color the triangle landed on it would determine the level of accuracy of the shot. As upgrades were purchased the green would expand, increasing the area to get a good shot. This could also be done on a gradient color scale. A bar like this is common in video games that have some form of launch function and this bar often represents power level.

Figure 4. An example of a graphical bar that could be used to allow the player to have some influence over their aim. The triangle would slide back and forth along the bar until the player clicked; the triangle would then stop in place, the color it lands on would indicate the accuracy of the shot. In this case, green would be most accurate and blue would be least accurate.

5 Conclusion

In conclusion, this Dart gun game shows that a game is a viable option to teach people more about statistical process control. But for this idea to be truly effective it needs to be refined and fine tune to best reflect the learning objectives while still being fun. It is a less traditional learning method than the classroom and the interactive aspect of it can help engage workers in the game and keep them focused on learning more about SPC.

References

[1] A. Ruiz-Martinez, F. Pereniguez-Garcia, R. Marin-Lopez, P. M. Ruiz-Martinez and A. F. Skarmeta-Gomez, "Teaching Advanced Concepts in Computer Networks: VNUML-UM Virtualization Tool," in *IEEE Transactions on Learning Technologies*, vol. 6, no. 1, pp. 85-96, Jan.-March 2013.

[2] Hunter, John. TEDTalks. "John Hunter: Teaching the World Peace Game". *Youtube*. Youtube, 26 Apr 2011. Web. 30 Sept 2016.

[3] Conversations with David Hardt and Jose Pacheco, Professors, MIT. Spring 2017.

[4] Wachs, A. "Do You Use SPC Correctly? Misapplication of Statistical Process Control Can Damage Your Operations." *Manufacturing Engineering*, vol. 134, no. 3, 01 Mar. 2005, pp. 159-168.

[5] Abel, Todd and Lisa Poling. "Hold My Calls: An Activity for Introducing Statistical Process." *Teaching Statistics*, vol. 37, no. 3, Sept 2015, pp. 96-103.

[6] "Training – Blue Agility." *Blue Agility*. N.p., n.d., Web. 15 May 2017.

[7] "Statistical Process Control (SPC)." *ASQ*. N.p., n.d. Web. 15 May 2017.

[8] "Lecture 44 – Statistical Process Control." *Coursera*. N.p., n.d. Web. 15 May 2017.

[9] Burroughs, B. "Facebook and FarmVille: A Digital Ritual Analysis of Social Gaming" *Games and Culture*, vol. 9, no. 3, 19 Jun. 2014.

[10] Conversations with Scot Osterweil, Professor, MIT. Fall 2016.

[11] Kumar, J and A. Marcus. "Gamification at Work: Designing Engaging Business Software." Design, User Experience, and Usability. Health, Learning, Playing, Cultural, and Cross-Cultural Use Experience Second International Conference, DUXU 2013 Held as Part of HCI International 2013, vol. 2, 01 Jan. 2013.

Appendix A

Python Code for Game – Designed by Suzanne Mueller

```
from tkinter import *

#Define global variables
# Default set up values for player
dartgun = "basic"
sight = "toilet paper"
location = "backyard"
shotsLeft = 10
points = 0


# Define values for canvas
canvas_width = 400
canvas_height = 400

# Define set up values for the target
radius = .5
num_rings = 7
x_center = canvas_width / 2
y_center = canvas_height / 2
ring_radius = (canvas_width / 3) / num_rings



# Shoot a dart at the target with mean distance and std_dev
def shootDart(r_mean, angle_mean):
        from math import pi, cos, sin
        from random import random
        # Angle std_dev is whole circle
        angle_std_dev = 2*pi
        r_std_dev = dartgunValues[dartgun][0] + locationValues[location][0] +
sightValues[sight][0]

        # Get the devs
        r_dev = r_std_dev * random()
        angle_dev = angle_std_dev * random()

        # Caluculate sum
        x = r_mean * cos(angle_mean) + r_dev * cos(angle_dev)
        y = r_mean * sin(angle_mean) + r_dev * sin(angle_dev)
```

```python
        return (x, y)

# Get the set up point (r, theta)
def getMeanR():
        from random import random
        # calculate the distance from center by adding the itemsValues held
        distance_by_item = dartgunValues[dartgun][0] + locationValues[location][0] +
sightValues[sight][0]
        # multiply the random number by the distance given due to items
        return random() * distance_by_item

def getMeanAngle():
        from random import random
        from math import pi
        # Choose a random angle between [0, 2pi)
        return random() * 2 * pi

def shootDarts(number):
        global shotsLeft
        print (number, shotsLeft)
        r_mean = getMeanR()
        angle_mean = getMeanAngle()
        for i in range(min(number, shotsLeft)):
                x, y = shootDart(r_mean, angle_mean)
                drawHit(x, y)
                shotsLeft -= 1

        updateShotsLeftVar()
        #check for end
        if(shotsLeft == 0):
                nextGameButton.config(state="normal")


def printBullseye():

        for i in range(num_rings, 0, -1):
                if i % 2 == 1:
                        color = "red"
                else:
                        color = "white"
                canvas.create_oval( x_center - ring_radius * i, y_center - ring_radius * i,
x_center + ring_radius * i, y_center + ring_radius * i, fill=color)

def drawHit(x, y):
        print ("DRAW HIT")
        x_center = canvas_width/2
        y_center = canvas_height /2
```

```python
        d = 5
        # update points
        if isHit(x,y):
                updatePoints(10) # the number of points you get for a hit

        # Convert to coordinates draw
        x = x*ring_radius / radius
        y = y * ring_radius / radius

        # Draw on board
        canvas.create_rectangle(x_center + x - d, y_center + y-d, x_center + x + d,
y_center + y + d, fill="green")


#arguments are actual coordinates
def isHit(x, y):
        from math import sqrt
        if sqrt(x*x + y*y) < 1: # Adjust this to change val
                return True
        return False

def updatePoints(inc):
        global points
        points += inc
        pointsVar.set("Points: $" + str(points))


def purchaseGun(val):
        global points, dartgun
        # Check if have enough points
        if points >= dartgunCost[val]:
                #switch to new gun
                dartgun = val
                #get rid of money
                updatePoints(- dartgunCost[val])
                #set value to 0 so can switch back
                dartgunCost[val] = 0
                updateCurrentStateVar()


def purchaseLocation(val):
        global points, location
        # Check if have enough points
        if points >= locationCost[val]:
                #switch to new location
                location = val
                #get rid of money
```

```python
                updatePoints(-locationCost[val])
                #set value to 0 so can switch back
                locationCost[val] = 0
                updateCurrentStateVar()

def purchaseSight(val):
        global points, sight
        # Check if have enough points
        if points >= sightCost[val]:
                #switch to new sight
                sight = val
                #get rid of money
                updatePoints(-sightCost[val])
                #set value to 0 so can switch back
                sightCost[val] = 0
                updateCurrentStateVar()

def takeShot():
        text = entry.get()
        num = 0
        try:
                num = int(text)
        except:
                num = 1
        shootDarts(num)
        canvas.update()

def resetGame(canvas):
        global shotsLeft
        drawCanvas(canvas)
        shotsLeft = 10
        updateShotsLeftVar()
        nextGameButton.config(state=DISABLED)

def drawCanvas(canvas):
        #color background
    canvas.create_rectangle(0,0, canvas_width, canvas_height, fill = "black")
        #draw bullseye
    printBullseye()

def setUpButtons(root):
    frameGuns = Frame(root)
    frameGuns.pack(side="top")
    Button(frameGuns, text = "basic" + ": $" + str(dartgunCost["basic"]), command =
lambda: purchaseGun("basic")).pack(side=LEFT)
    Button(frameGuns, text = "midrange" + ": $" + str(dartgunCost["midrange"]),
command = lambda: purchaseGun("midrange")).pack(side=LEFT)
```

```python
    Button(frameGuns, text = "superior 1" + ": $" + str(dartgunCost["superior 1"]),
command = lambda: purchaseGun("superior 1")).pack(side=LEFT)
    Button(frameGuns, text = "superior 2" + ": $" + str(dartgunCost["superior 2"]),
command = lambda: purchaseGun("superior 2")).pack(side=LEFT)


    frameSight = Frame(root)
    frameSight.pack(side="top")
    Button(frameSight, text = "toilet paper" + ": $" + str(sightCost["toilet paper"]),
command = lambda: purchaseSight("toilet paper")).pack(side = LEFT)
    Button(frameSight, text = "red dot" + ": $" + str(sightCost["red dot"]), command =
lambda: purchaseSight("red dot")).pack(side = LEFT)
    Button(frameSight, text = "laser" + ": $" + str(sightCost["laser"]), command =
lambda: purchaseSight("laser")).pack(side = LEFT)


    frameLoc = Frame(root)
    frameLoc.pack(side="top")
    Button(frameLoc, text = "backyard" + ": $" + str(locationCost["backyard"]), command
= lambda: purchaseLocation("backyard")).pack(side=LEFT)
    Button(frameLoc, text = "outdoor" + ": $" + str(locationCost["outdoor"]), command =
lambda: purchaseLocation("outdoor")).pack(side=LEFT)
    Button(frameLoc, text = "indoor" + ": $" + str(locationCost["indoor"]), command =
lambda: purchaseLocation("indoor")).pack(side=LEFT)


def updateCurrentStateVar():
        currentStateVar.set("Dartgun: " + dartgun + ", Location: " + location + ", Sight: "
+ sight)

def updateShotsLeftVar():
        shotsLeftVar.set("Shots Left: " + str(shotsLeft))

# Set up the canvas of the dart game
def setUp():
  global root, canvas, entry, pointsText, shotsText, nextGameButton, pointsVar,
shotsLeftVar, currentStateVar
  root = Tk()
  root.title("Darts: Thesis Game")
  canvas = Canvas(root,width = canvas_width, height = canvas_height, bg = 'black')
  canvas.pack(expand = YES, fill = BOTH)

  #draw background
  drawCanvas(canvas)
  #Insert Text
  currentStateVar = StringVar()
  updateCurrentStateVar()
```

```python
    currentText = Label(root, textvariable = currentStateVar)
    currentText.pack(expand= YES, fill = BOTH)

    pointsVar = StringVar()
    updatePoints(0)
    pointsText = Label(root, textvariable = pointsVar)
    pointsText.pack(expand= YES, fill = BOTH)

    shotsLeftVar = StringVar()
    updateShotsLeftVar()
    shotsLeft = Label(root, textvariable = shotsLeftVar)
    shotsLeft.pack(expand= YES, fill = BOTH)


    # Set up an text box to enter numbers of desired shots
    entry = Entry(root)
    entry.pack(expand= YES, fill = BOTH)

    button = Button(root, text="Shoot", command=takeShot)
    button.pack(expand= YES, fill = BOTH)

    setUpButtons(root)

    nextGameButton = Button(root, text="Next Game", command=lambda:
resetGame(canvas), state=DISABLED)
    nextGameButton.pack(expand= YES, fill = BOTH)

    return canvas, root


# Define the cost of the guns
dartgunCost =          {
        "basic": 0,
        "midrange": 100,
        "superior 1": 250,
        "superior 2": 200
}

locationCost = {
        "backyard": 0,
        "outdoor": 50,
        "indoor": 200
}

sightCost =   {
        "toilet paper": 0,
        "red dot": 150,
```

```python
        "laser": 250
}

# Define the mean and std_dev with each item
dartgunValues = {
        "basic": (1.0, 0.9),
        "midrange": (0.5, 0.65),
        "superior 1": (0.2, 0.35),
        "superior 2": (0.3, 0.25)
}

locationValues = {
        "backyard": (1.2, 1.25),
        "outdoor": (0.7, 1.25),
        "indoor": (0.7, 0.6)
}

sightValues =            { # (mean, std_dev)
        "toilet paper": (1.0, 0.0),
        "red dot": (0.5, 0.0),
        "laser": (0.1, 0.0)
}


def main():
        canvas, root = setUp()
        root.mainloop();

if __name__ == "__main__": main()
```

Appendix B

Pre-Test Questions

Name:

University:

Course/Major(s):

1) Have you taken 2.008 or another Manufacturing class (note: 2.007 does not count) before?

2) Have you taken a statistics class at your university before such as 18.05 for MIT?

3) In a sentence or two, what is Statistical Process Control?

Frank shoots a target 4 times with a bow and arrow. Assuming Frank aimed for the middle and that he controlled all possible variables before shooting, which of the four targets are shown below is most likely to be Frank's. Please write a brief explanation below the targets for why you think what you selected is correct.

a)

b)

c)

d)
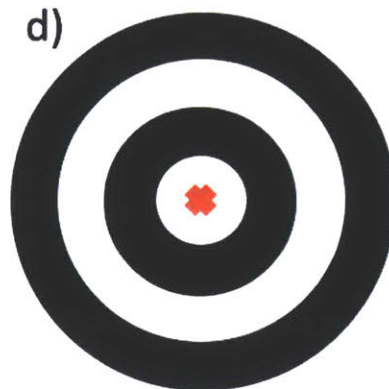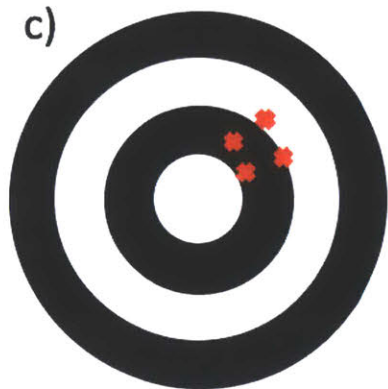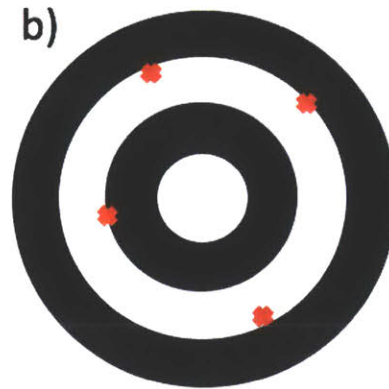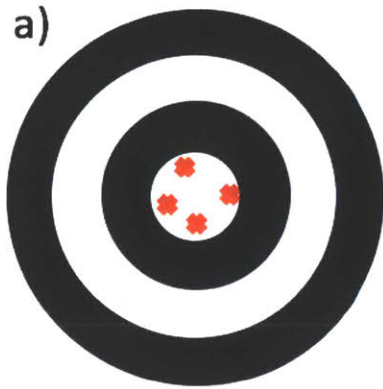
4) At the start of the day, a few injection molded parts are sampled and the following curve is found:

-0.5   0.5   1.5   2.5   3.5   4.5   5.5   6.5
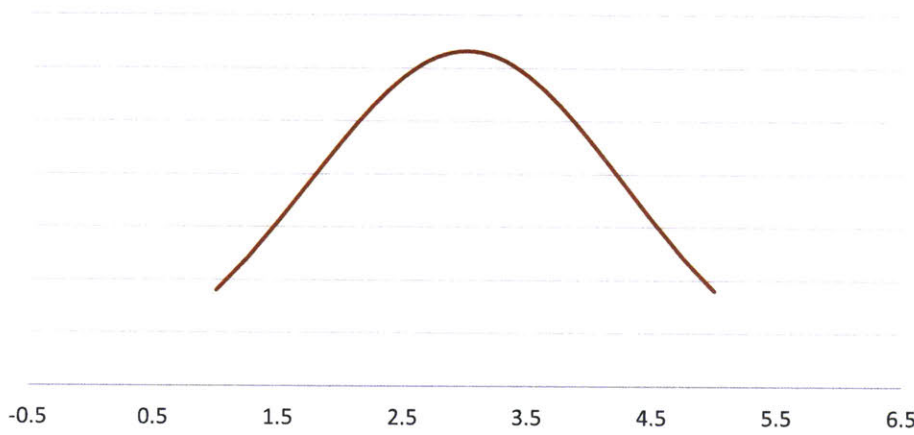
Later in the afternoon, a few more parts are tested. You notice the variability of these parts have changed. Which graph shows this change in variability?

-0.5   0.5   1.5   2.5   3.5   4.5   5.5   6.5

a)

-0.5   0.5   1.5   2.5   3.5   4.5   5.5   6.5

b)
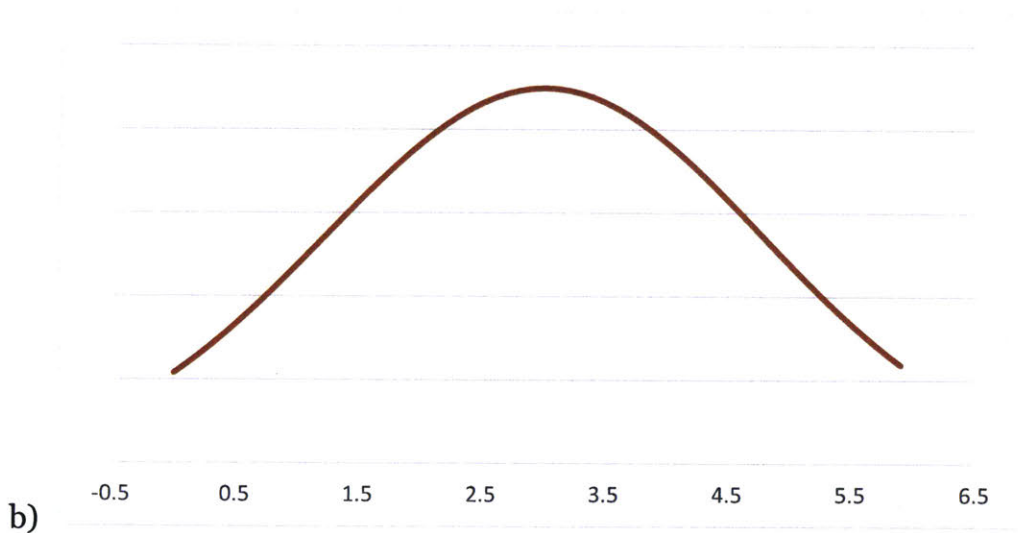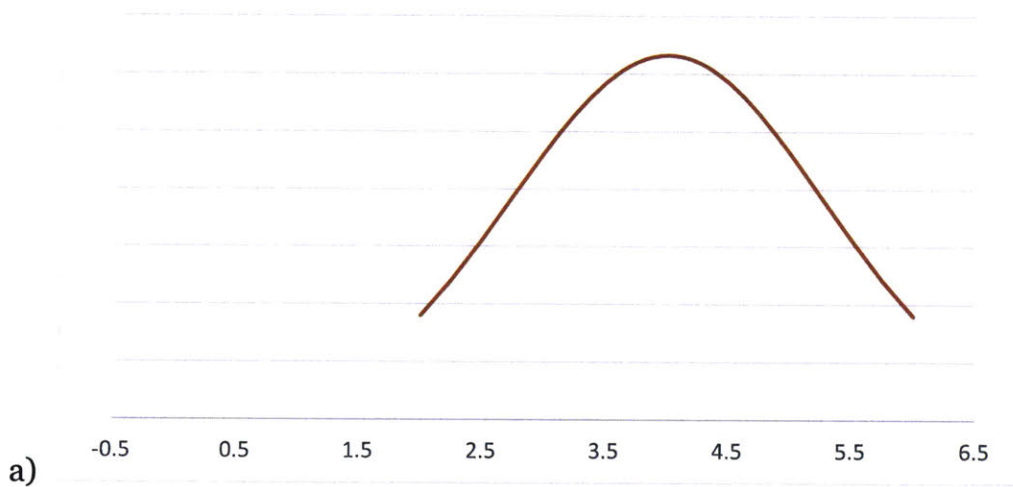
5) Which of the following is a reason that could explain this change in variance?
    a. The machine had heated up since the morning and the temperature difference was impacting the parts.
    b. A new shipment of plastic was being used and the chemical make-up was slightly different.
    c. The molds for this part had reached the end of their life cycle and no longer lined up perfectly every time.
    d. All of the above.

Post-Test Questions

1) Plots a, b, and c are shown below.

(a)
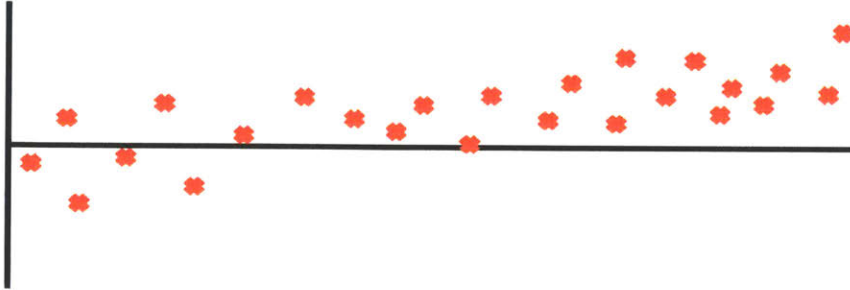
(b)
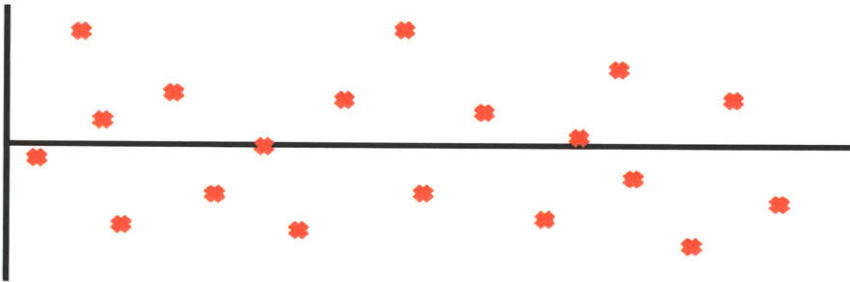
(c)
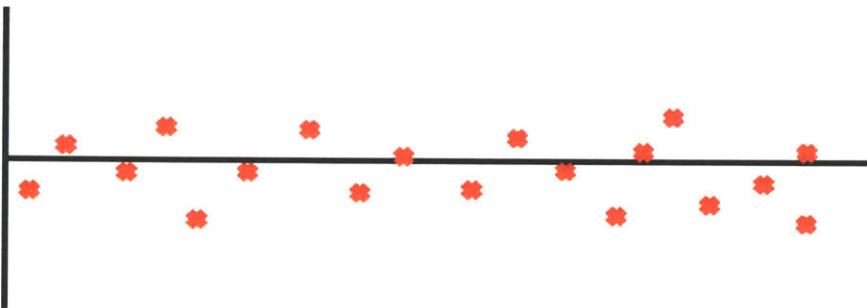
    i)   Which graph has the most variability?

    ii)   Which graph represents a shift in the mean?

    iii)   Which graph represents a controlled random process?

1) In game, when you replaced your sight with a new and improved one, what happened to the distribution?

   a. Standard Deviation Decreased
   b. Variance Increased
   c. The Mean Shifted
   d. All of the Above
   e. None of the above

2) In game, when you located from the outdoor shooting range to the indoor shooting range, what happened to the distribution?

   a. Standard Deviation Decreased
   b. Variance Decreased
   c. Mean Was Unaffected
   d. All of the Above
   e. None of the above

Appendix C

Pre-Test and Post-Test Results

There were 12 participants. 7 were Mechanical Engineering Students, 4 had taken a manufacturing class, and 3 had taken a statistics class. One student had taken both a manufacturing and a statistics class.

	Score Increase	Score Decrease	Same Score	Total
Total	5	4	3	12
Mechanical Engineering Student	3	2	2	7
Manufacturing Class	1	1	2	4
Statistics Class	0	2	1	3

Above is a summary of results from the participants based on their major and classes compared to whether their score increased, decreased, or remained the same.

	Pre-Test		
	Question 1	Question 2	Question 3
Total	10	9	10
Mechanical Engineering Student	6	5	6
Manufacturing Class	3	3	4
Statistics Class	3	3	3

	Post-Test		
	Question 1	Question 2	Question 3
Total	12	10	7
Mechanical Engineering Student	7	6	4
Manufacturing Class	4	3	2
Statistics Class	3	1	2

Above is the breakdown of correct answers for each question asked in the pre-test and post-test.

Slow and Steady Shooter – 25 Rounds

Dartgun: basic, Location: backyard, Sight: toilet paper
Points: $30

Dartgun: basic, Location: backyard, Sight: toilet paper
Points: $60
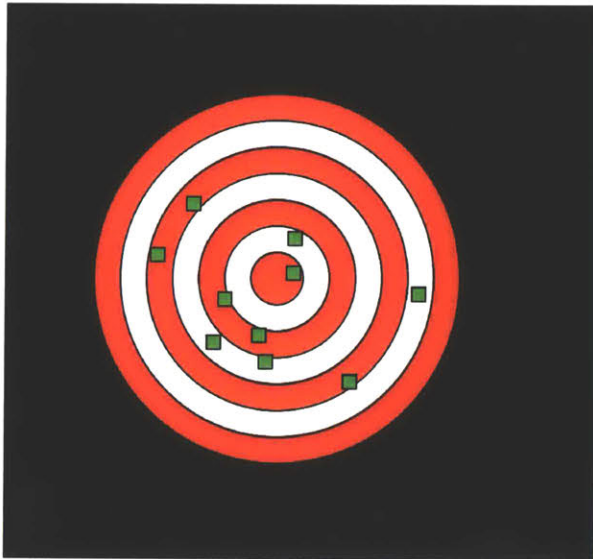
Dartgun: basic, Location: outdoor, Sight: toilet paper
Points: $30

Dartgun: basic, Location: outdoor, Sight: toilet paper
Points: $70
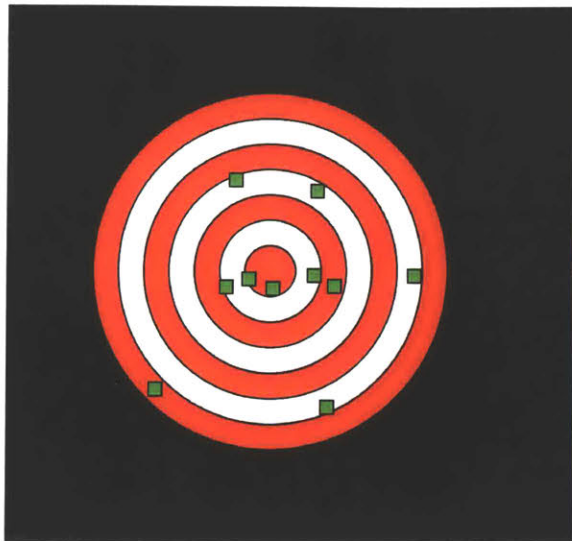
Dartgun: basic, Location: outdoor, Sight: toilet paper
Points: $110
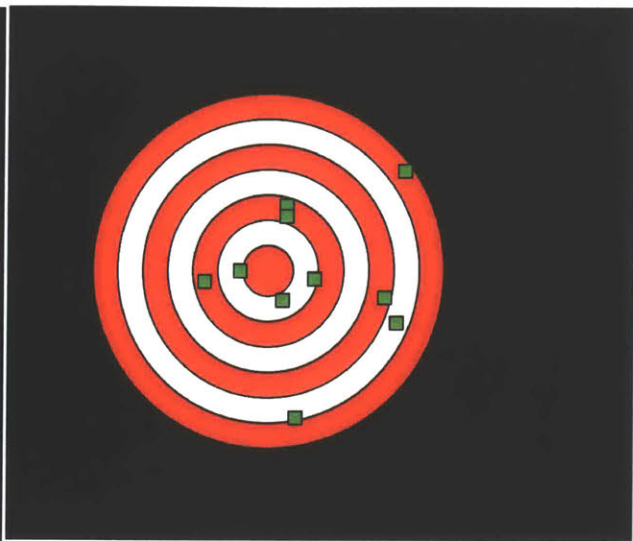
Dartgun: midrange, Location: outdoor, Sight: toilet paper
Points: $40
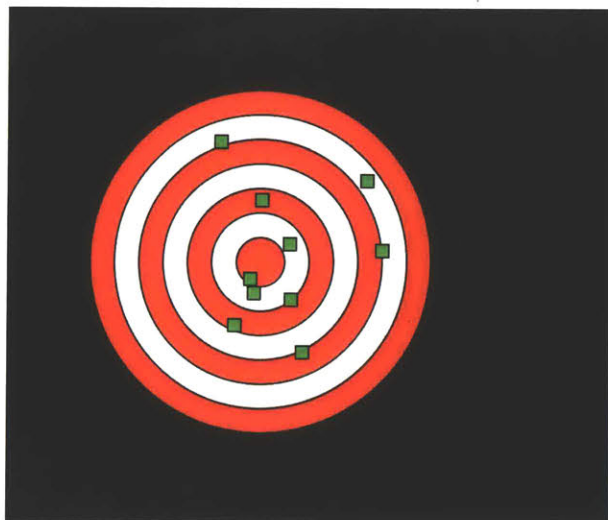
Dartgun: midrange, Location: outdoor, Sight: toilet paper
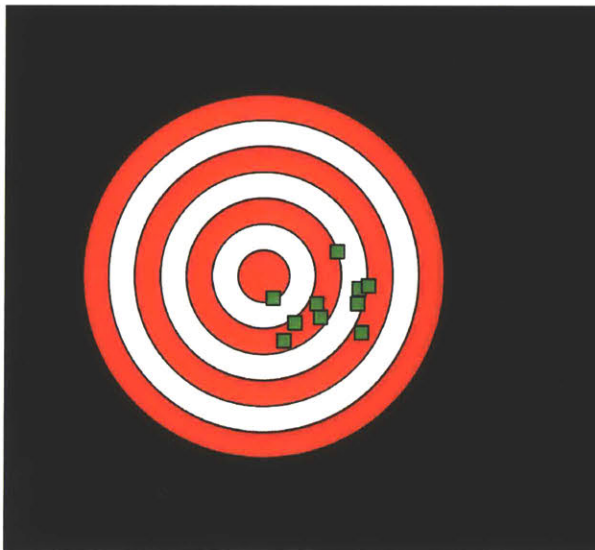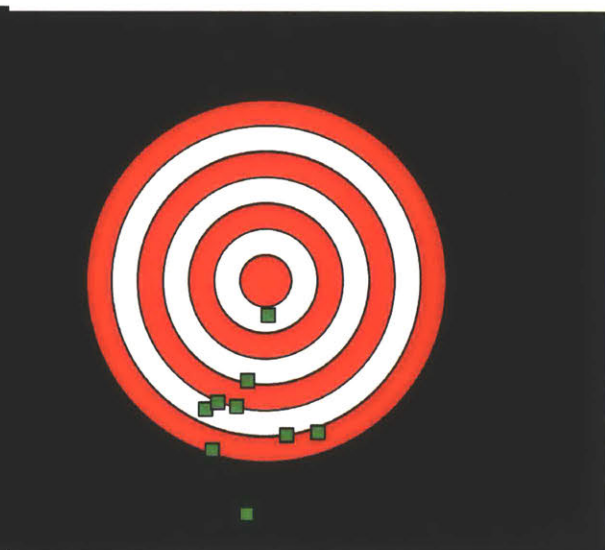Points: $80
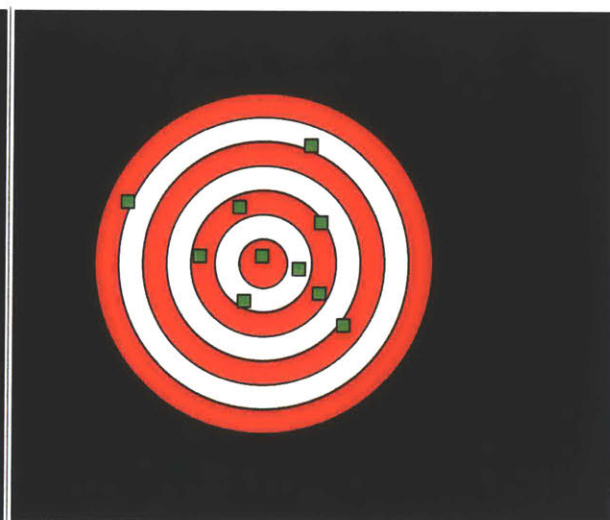
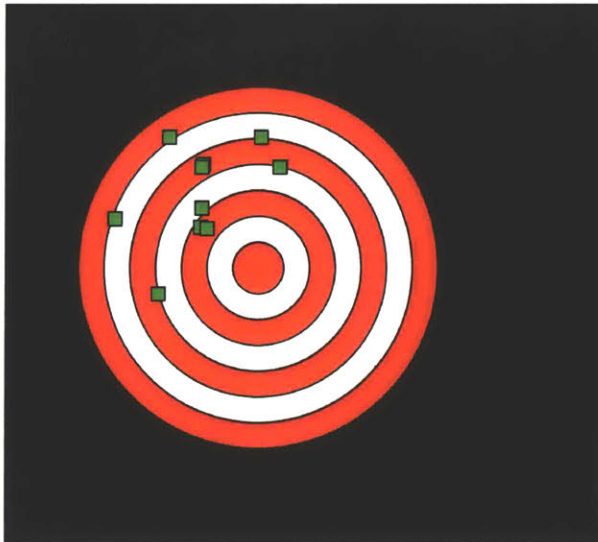Dartgun: midrange, Location: outdoor, Sight: toilet paper
Points: $110
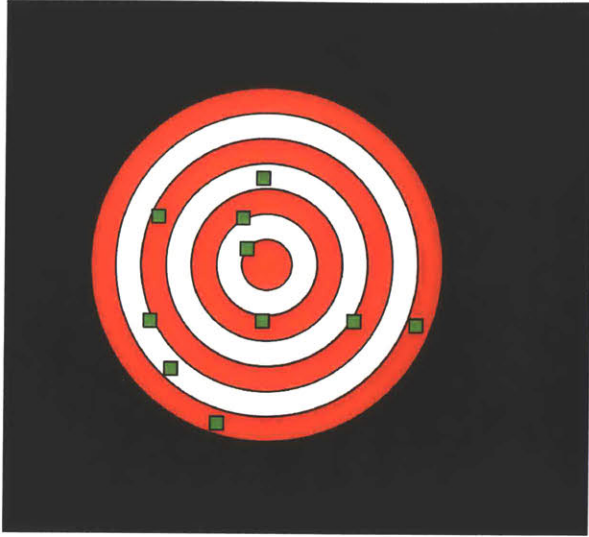
Dartgun: midrange, Location: outdoor, Sight: toilet paper
Points: $120
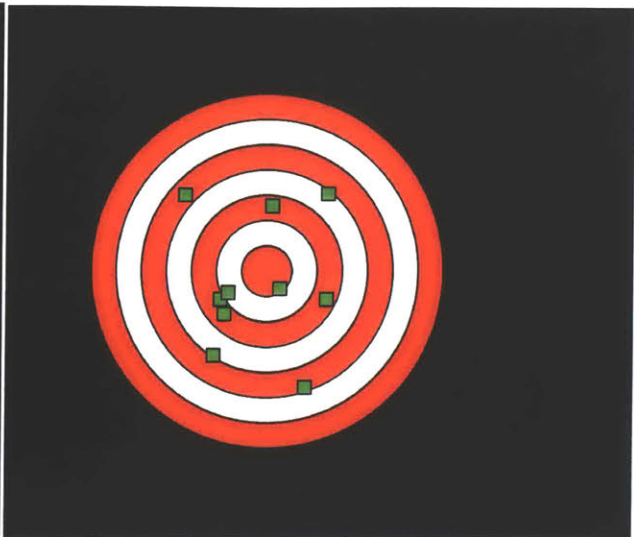
Dartgun: midrange, Location: outdoor, Sight: toilet paper
Points: $140
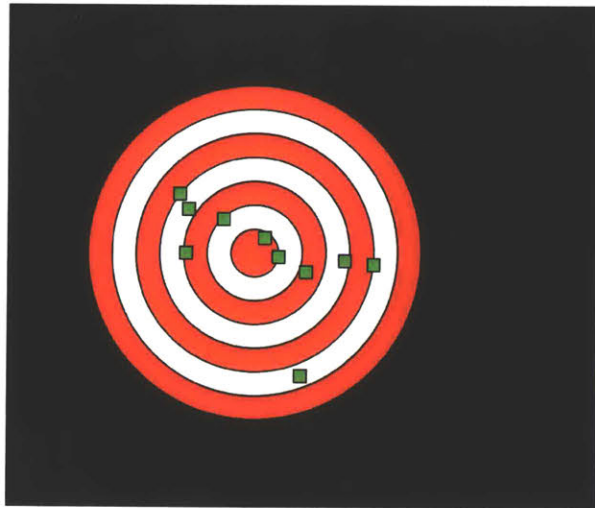
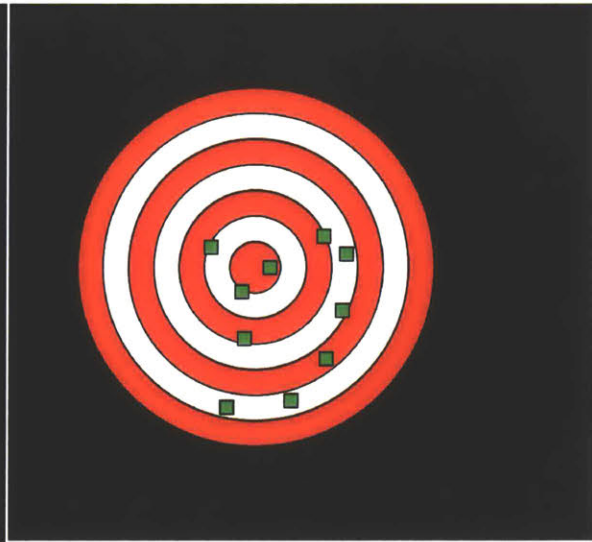Dartgun: midrange, Location: outdoor, Sight: toilet paper
Points: $170
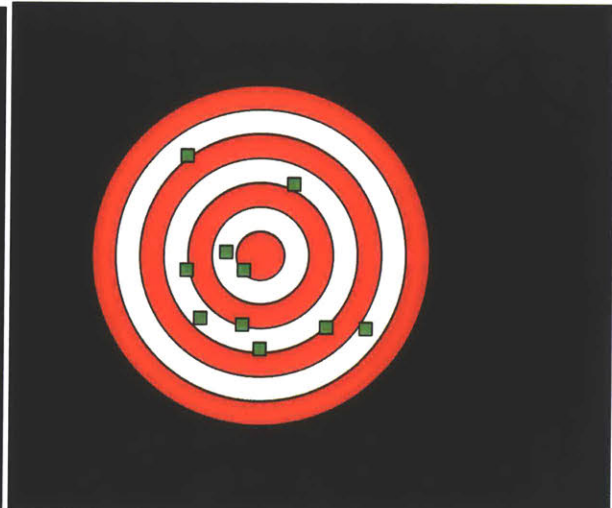
Dartgun: midrange, Location: outdoor, Sight: red dot
Points: $40
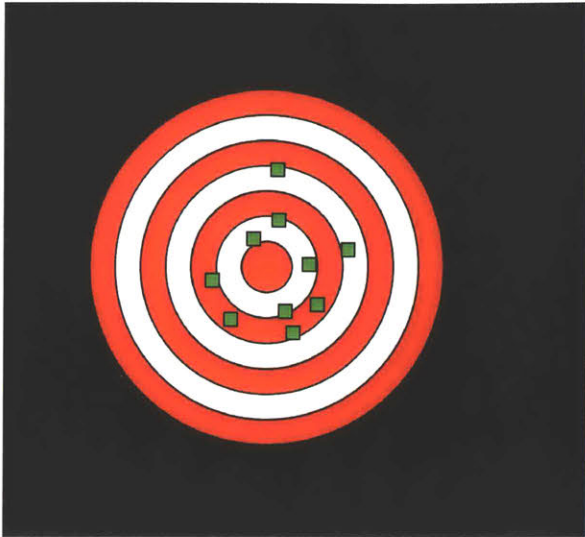
Dartgun: midrange, Location: outdoor, Sight: red dot
Points: $80
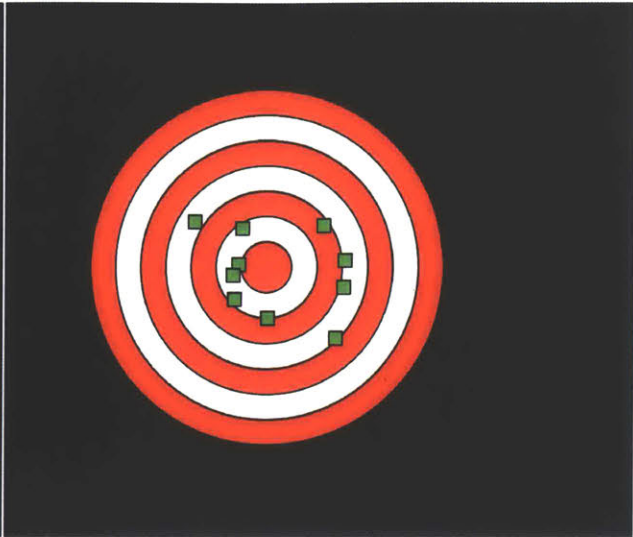
Dartgun: midrange, Location: outdoor, Sight: red dot
Points: $130
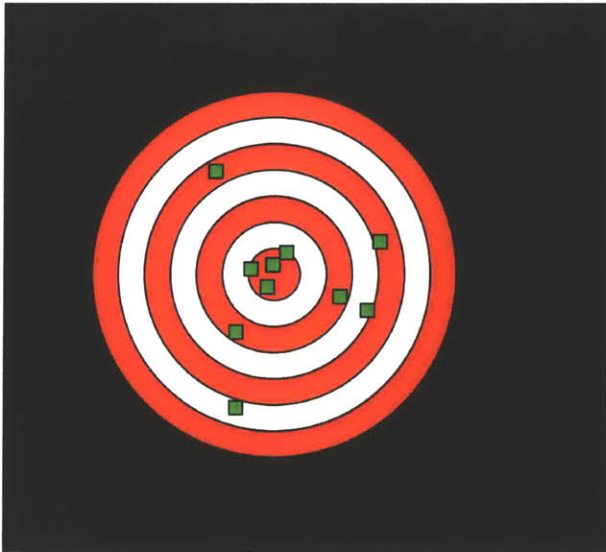
Dartgun: midrange, Location: outdoor, Sight: red dot
Points: $170
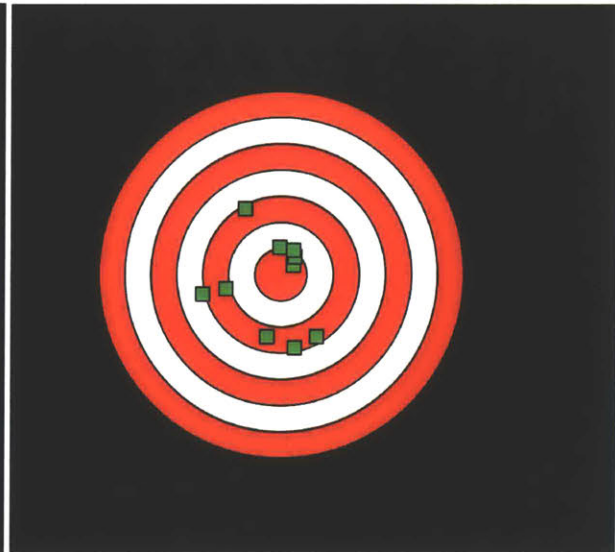
Dartgun: midrange, Location: outdoor, Sight: red dot
Points: $210

Dartgun: midrange, Location: indoor, Sight: red dot
Points: $40
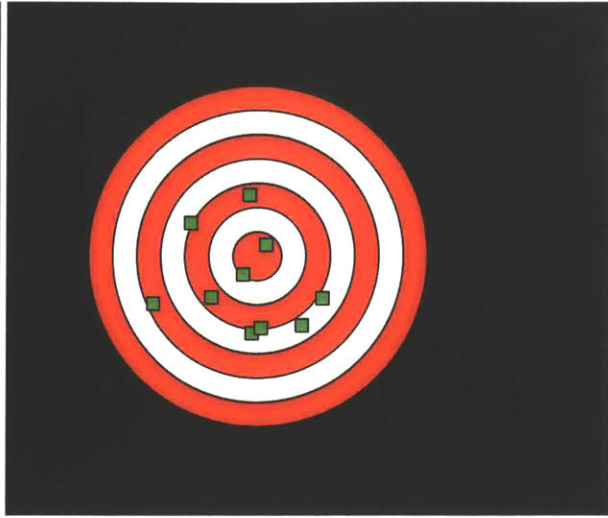
Dartgun: midrange, Location: indoor, Sight: red dot
Points: $60
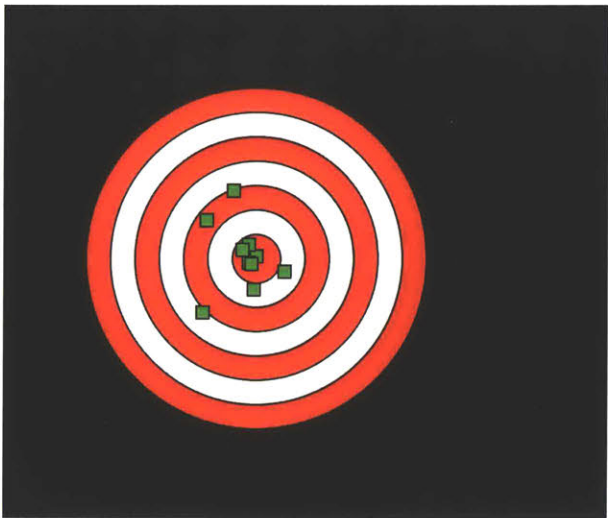
Dartgun: midrange, Location: indoor, Sight: red dot
Points: $130
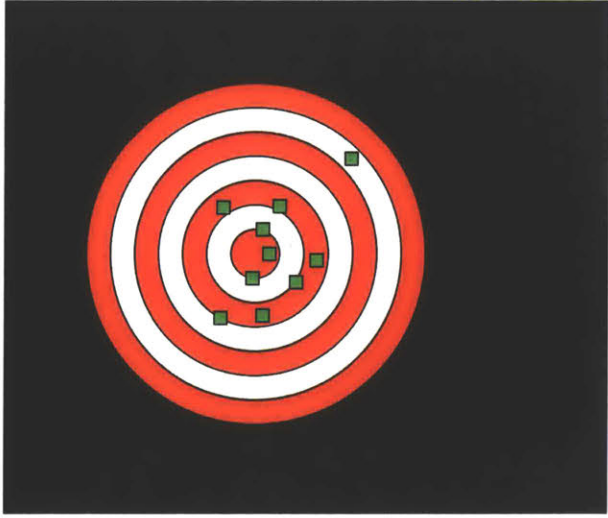
Dartgun: midrange, Location: indoor, Sight: red dot
Points: $160
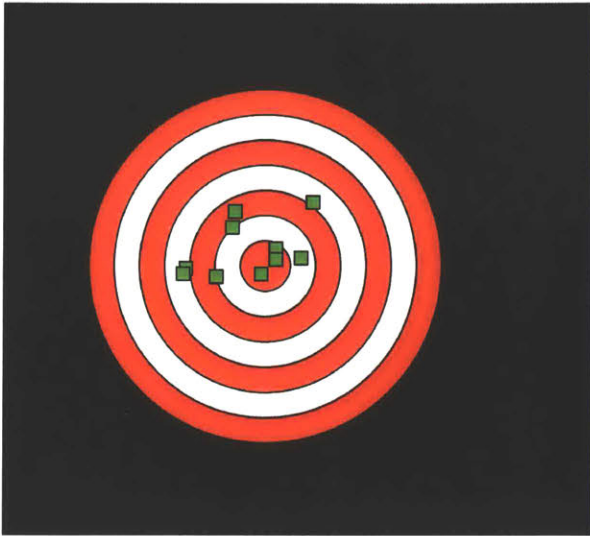
Dartgun: midrange, Location: indoor, Sight: red dot
Points: $200
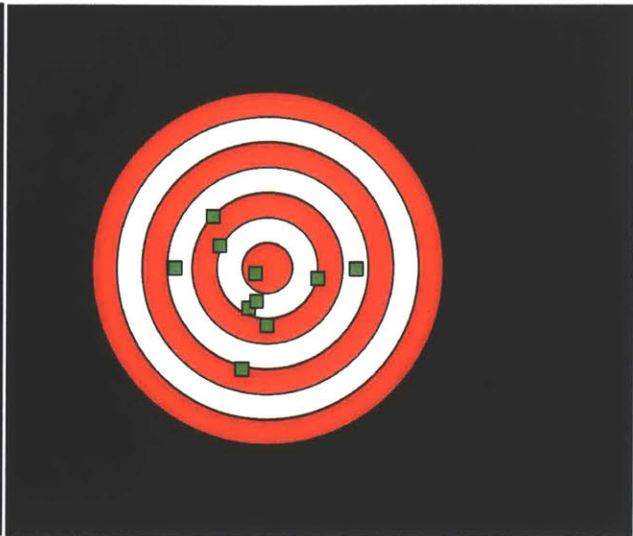
Dartgun: midrange, Location: indoor, Sight: red dot
Points: $40
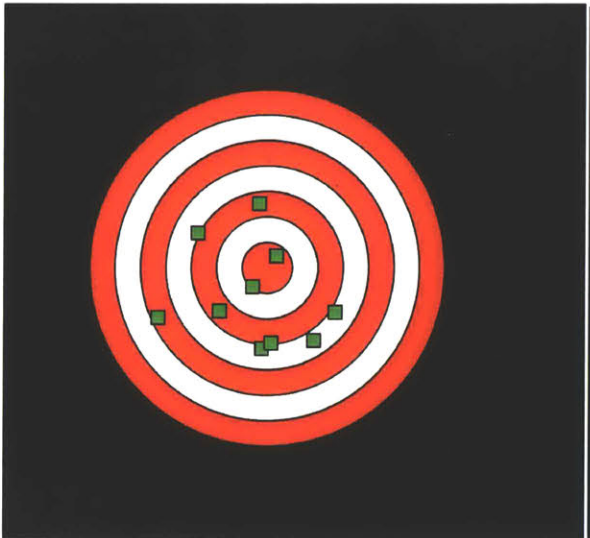
Dartgun: midrange, Location: indoor, Sight: red dot
Points: $60

Dartgun: midrange, Location: indoor, Sight: red dot
Points: $130

Dartgun: midrange, Location: indoor, Sight: red dot
Points: $160

Appendix E

Rapid Fire Shooter – 25 Rounds
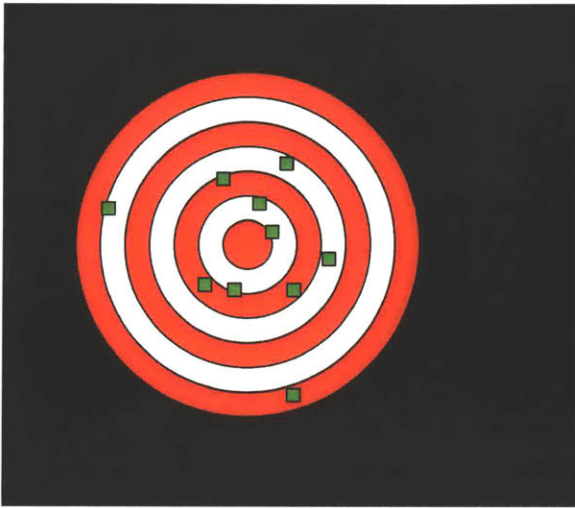
Dartgun: basic, Location: backyard, Sight: toilet paper
Points: $30

Dartgun: basic, Location: backyard, Sight: toilet paper
Points: $30

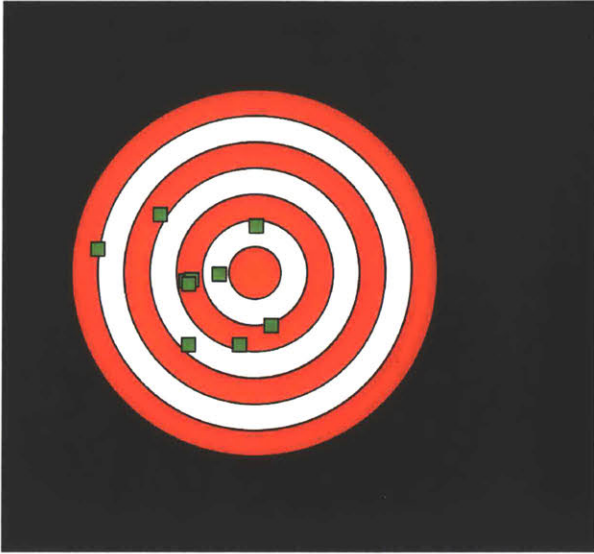Dartgun: basic, Location: backyard, Sight: toilet paper
Points: $40

Dartgun: basic, Location: backyard, Sight: toilet paper
Points: $50

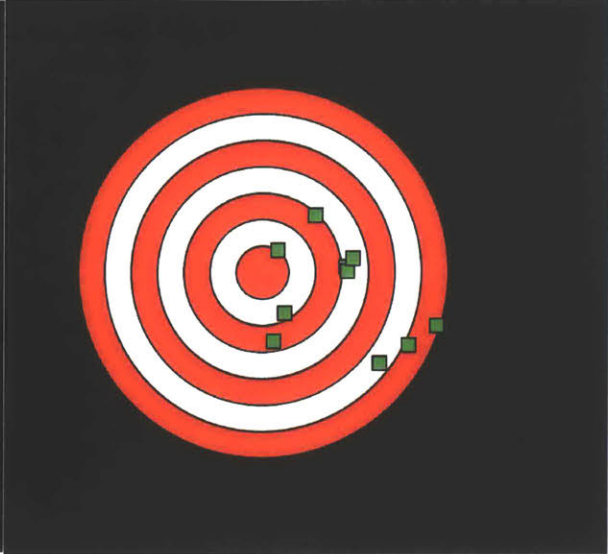Dartgun: basic, Location: outdoor, Sight: toilet paper
Points: $20

Dartgun: basic, Location: outdoor, Sight: toilet paper
Points: $20

Dartgun: basic, Location: outdoor, Sight: toilet paper
Points: $20

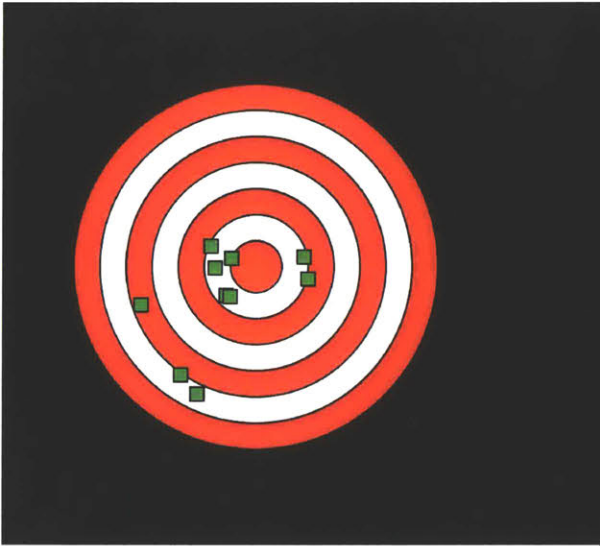Dartgun: basic, Location: outdoor, Sight: toilet paper
Points: $50

Dartgun: basic, Location: outdoor, Sight: toilet paper
Points: $110

Dartgun: midrange, Location: outdoor, Sight: toilet paper
Points: $10

Dartgun: midrange, Location: outdoor, Sight: toilet paper
Points: $10

Dartgun: midrange, Location: outdoor, Sight: toilet paper
Points: $40
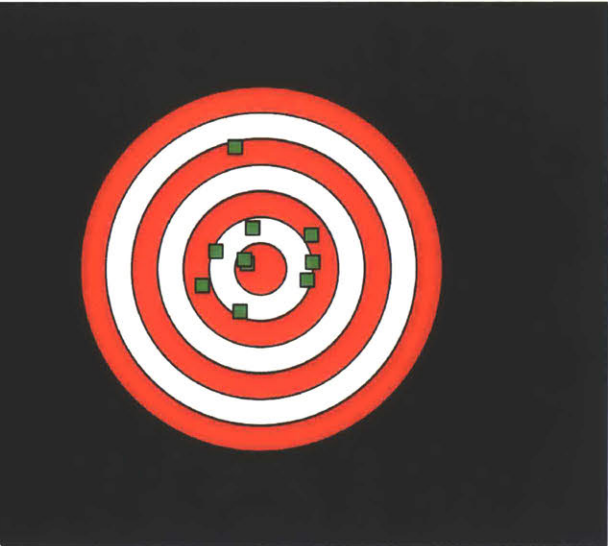
Dartgun: midrange, Location: outdoor, Sight: toilet paper
Points: $60

Dartgun: midrange, Location: outdoor, Sight: toilet paper
Points: $80

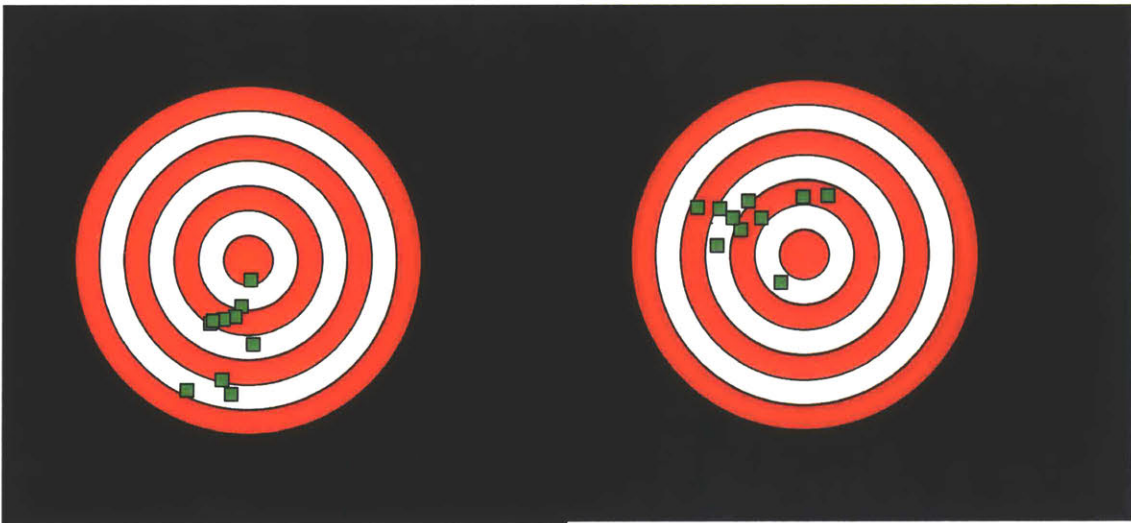Dartgun: midrange, Location: outdoor, Sight: toilet paper
Points: $140

Dartgun: midrange, Location: outdoor, Sight: toilet paper
Points: $210
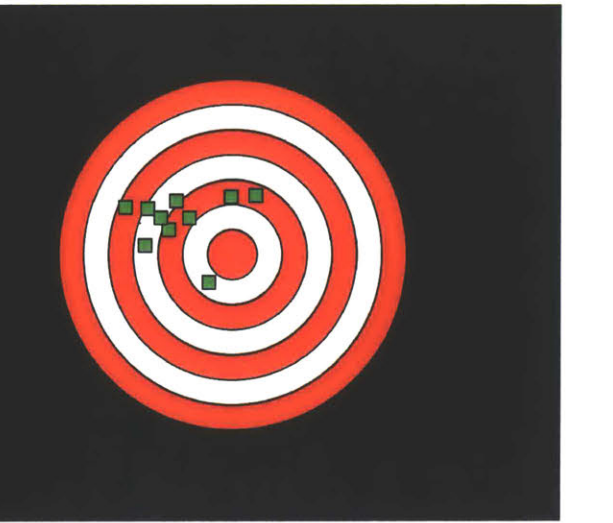
Dartgun: midrange, Location: outdoor, Sight: red dot
Points: $100

Dartgun: midrange, Location: outdoor, Sight: red dot
Points: $160

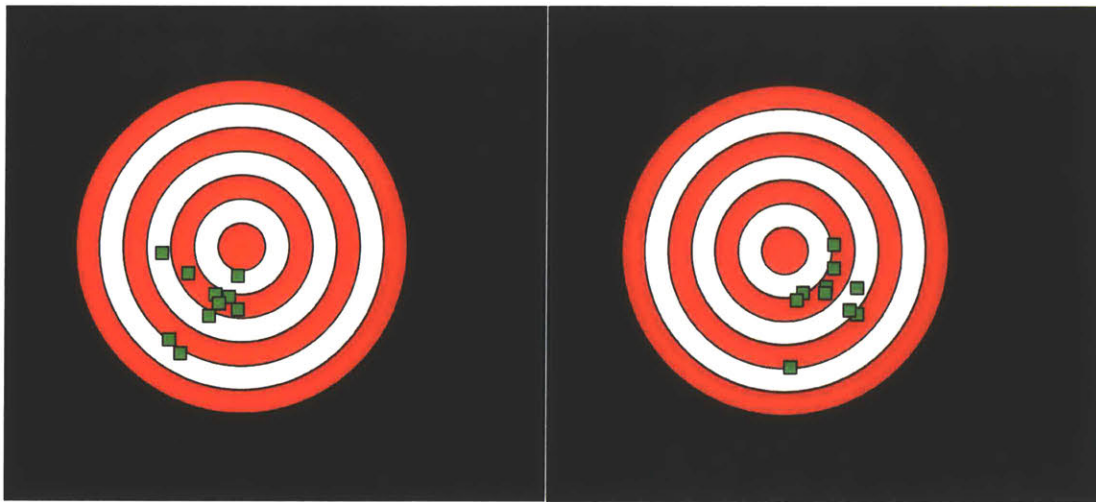Dartgun: midrange, Location: outdoor, Sight: red dot
Points: $180

Dartgun: midrange, Location: outdoor, Sight: red dot
Points: $190

Dartgun: midrange, Location: outdoor, Sight: red dot
Points: $240

Dartgun: midrange, Location: indoor, Sight: red dot
Points: $100

Dartgun: midrange, Location: indoor, Sight: red dot
Points: $110

Dartgun: midrange, Location: indoor, Sight: red dot
Points: $120

Dartgun: midrange, Location: indoor, Sight: red dot
Points: $160