Installing Python Project

1. Installing Python
    a. Download the Python 3.7 installer from here.
    b. Double click on the installer. Choose the "Install Now" option and continue following the instructions.
    c. The installation will come with Pip, which is tool for installing Python packages.
2. Installing Pygame
    a. Download the correct pygame installer from here, based on your operating system.
    b. Open a command prompt
    c. Install pygame with the following Pip command: <pip directory> install <pygame installer directory>. E.g. C:\Python37\Scripts\pip install C:\downloads\pygame-1.9.4-cp37-cp37m-win_amd64.whl
3. Running the project
    a. Download project master branch from git repository (https://github.com/ga6198/CSC4992Project.git).
    b. Click on your computer's search bar in the bottom left corner, and search for "IDLE".
    c. Open Python IDLE.
    d. Click File>Open, find the CSC4992Project directory, and open the main.py file by right-clicking and choosing "Edit with IDLE".
    e. Click Run>Run Module to run the main.py file, which starts the game. NOTE: Make sure your current working directory is inside the project folder, or else the game will not start correctly.
    f. Type "main", "single", or "multi" to directly jump to the respective pages, or "quit" to close the project.

User Manual

Game Description

Welcome to the Python Memory Game!

In Memory, several cards are displayed face down on a board. Each card has a matching pair. The goal of a single player game is to match pairs until all cards on the board are matched. Players will flip up two cards at a time. If the cards do not match, they will flip back over. Be sure to remember their positions! If the cards do match, then points are added to the score, the cards will disappear, and the score multiplier is increased. The score multiplier helps add bonus points onto your score and will increase for every pair you match in a row. However, if you lose your matching streak, the multiplier will reset. Try to get the highest score possible! After a single player game, you can save your score, as well as view other people's scores.

In multiplayer, the rules are the same as a single player game, but players will take turns matching two cards at a time. The player with the highest score at the end wins the round.

The following parts of this document will help you navigate through each menu.

Starting Terminal

1. Opening main.py opens up a Python terminal.
2. Type one of the given options to directly jump to a page.
    a. Type "main" to go to the main page.
    b. Type "single" to go to the single player page.
    c. Type "multi" to go to the multiplayer page.
    d. Type "quit" to close the project.
3. Game display will open in background.

Main Menu

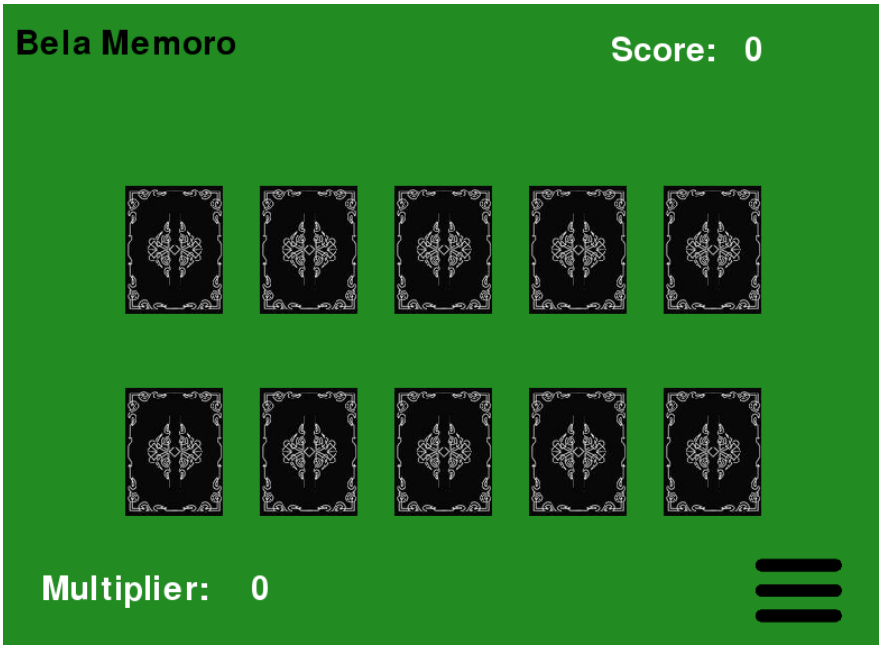
1. Single Player Button: Choose for a single player game. Takes you to the Single Player Menu.

2. Multiplayer Button: Choose for a multiplayer game. Takes you to the Multiplayer Menu.
3. Options Button: Adjust options. Takes you to the Options Menu.

Single Player Menu

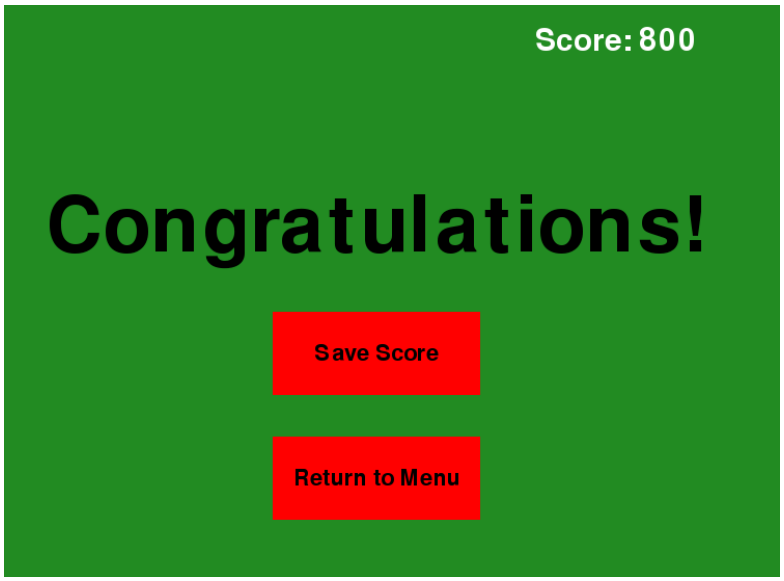
1. Back Button: Go back to the previous page. Takes you to the Main Menu.
2. Easy Button: Start a game with 10 cards. Takes you to Gameplay Menu (Single Player).
3. Medium Button: Start a game with 14 cards. Takes you to Gameplay Menu (Single Player).
4. Hard Button: Start a game with 18 cards. Takes you to Gameplay Menu (Single Player).
5. Options Button: Adjust options. Takes you to the Options Menu.

Gameplay Menu (Single Player)



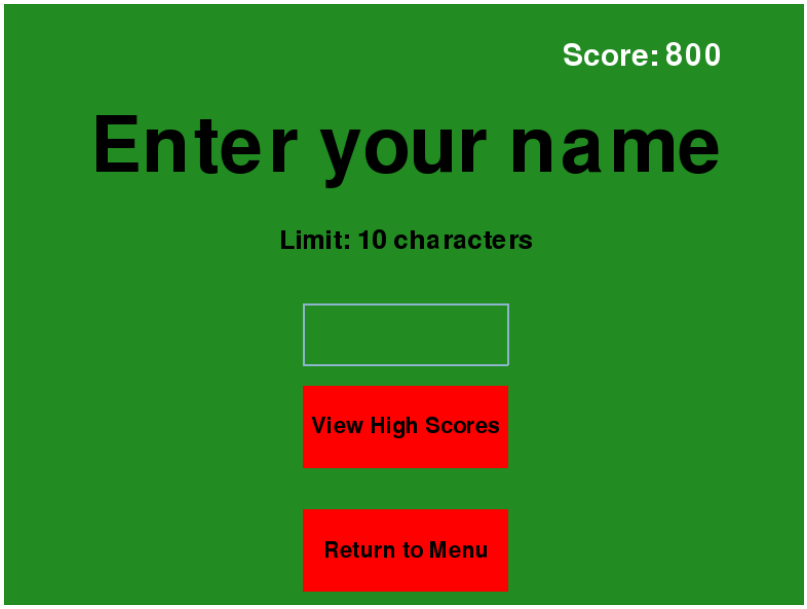
1. Back Button: Go back to the previous page. Takes you to the Single Player Menu.
2. Cards: Click on a card to flip it. When two cards are flipped, their images are compared. If they match, they clear, but if they don't match, they flip back down. If all cards are matched, you are immediately sent to the Results/Congratulations Menu.
3. Score: Current score.
4. Multiplier: Current score multiplier.
5. Options Button: Adjust options. Takes you to the Options Menu.

Results/Congratulations Menu



1. Save Score Button: Allows you to save your game score. Takes you to the Enter Your Name Menu.
2. Return to Menu Button: Takes you to the Main Menu.

Enter Your Name Menu



1. Input Box: Type your name (limit of 10 characters), and press the “enter” key to save your name and score. Takes you to the High Score Menu.
2. View High Scores Button: Click to view all high scores for the current card setting. Takes you to the High Scores Menu.
3. Return to Menu Button: Takes you to the Main Menu.

High Scores Menu

1. Scores: A list of top 10 high scores over the lifetime of the game. If there are less than 10 scores, the remaining entries are filled with “N/A.”
2. Return to Menu Button: Takes you to the Main Menu.
3. Test Recursion Button: Shows the use of recursion in the project. Sorts list of scores by name, asks for a name **in the terminal**, and returns the index of the name through binary search. After the task is finished, control returns to the game display.

Project proposal

Team Name: Bela Memoro

Names:	**Access IDs**:
1) Somali Bayi	ei1757
2) Ibrahim Hakim	fvi349
3) Nicole Racovites	er8015
4) Kevin Zhang	ga6198

Project Proposal Questionnaire:

1. Describe your project in one paragraph.

Our project is a simple Memory Card Game. The user has the option of a single or a multiplayer game. Then they have the option of what level of the game they would like to play (easy, medium, hard). They clicks on two cards to see if they match according to their pictures. If they are a match they get points, if they do not match they are flipped back over. The user goes through all the cards until the game is over.

2. Did you include a prototype presentation file into your 'Design' folder in your project repository?

3. What type of project are you doing (CRUD – Game – Data science – etc)?

Our project is a game that has both one player and multiplayer options for the user.

4. What is the URL of your repository?

https://github.com/ga6198/CSC4992Project.git

5. Did you include your instructor and TA in your repository?

Yes, both the instructor and TA is in our repository master.

6. Did you include the following files in a folder named 'Management' in your project repository?

- Team contract
- Planning document
- Gantt Chart (basic and advanced features clearly annotated)

Yes, we have the planning document and Gantt chart in our management folder.

7. Did you add your tasks to your project management tool? Which tool are you using? E.g. Project in Github

We used the Kanban board to add and move tasks for our program. The board was an easy tool that made it more efficient to keep track of what needed to be done.

8. How familiar are your team members with OOP and UML diagrams?

We are familiar with the Object-Oriented Program and UML diagrams.

Project Outline

1. Names and access ID of students that participated in the elaboration of this report.

ga6198 Kevin Zhang

er8015 Nicole Racovites

ei1757 Somali Bayi

fvi349 Ibrahim Hakim

2. Describe the progress that you have done so far. Some questions that you can answer include:

- Did you tested any of the open source projects written in Python that are similar to your project? If so, What did you learn from the projects that you found? Is there any project that have already implemented part of what you want to do? Did you decided to contribute to an open source project instead of doing your project for scratch? If so, explain all the details of the project and your anticipated contribution. Include this information in your repository as well.

Our process in our group project is going quite well. We have all been communicating through the GroupMe, which through there we decide if we want to meet after class or if we cannot make a meet-up session. We use GitHub to distribute and access code easily. We have done research on some open source projects yet we have been coding from scratch. Although, the open source projects have helped tremendously on where we wanted our project to go.

- Did you try any python framework? Describe the process and screenshots of your demo.

Yes we tried python framework. Below is screenshots of our demo project

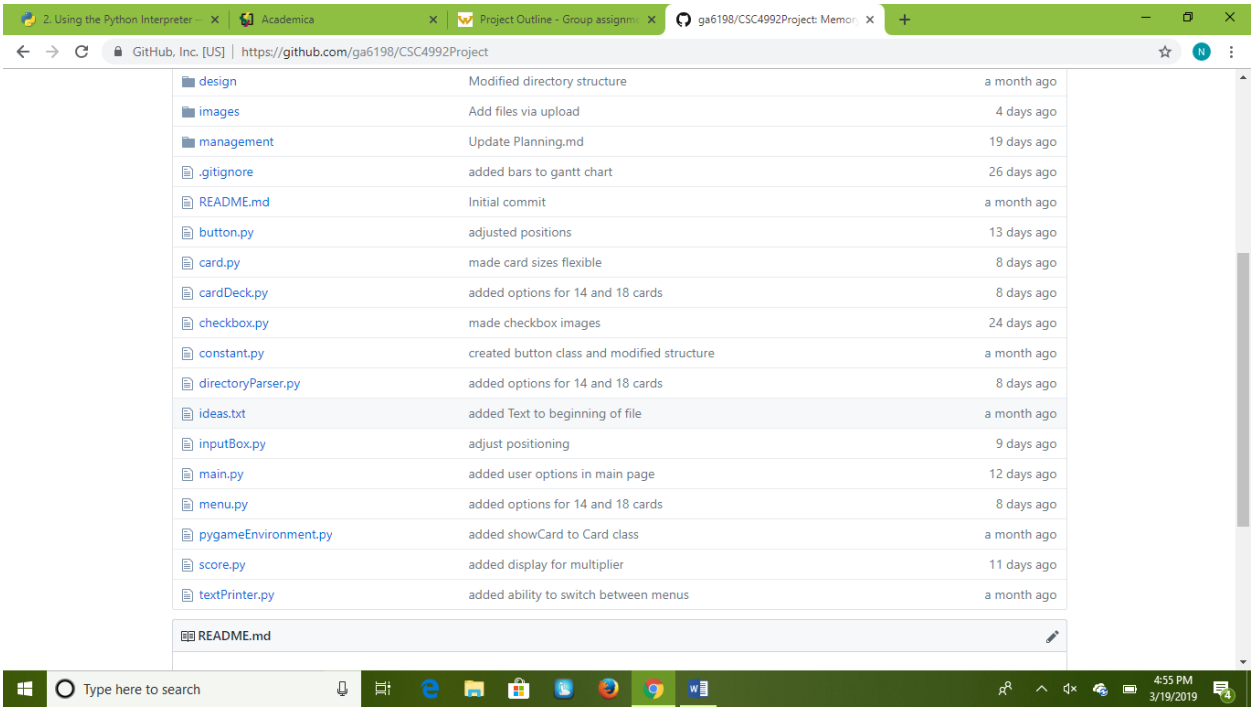
- Have you tried to run a demo or framework and you got stuck? Describe the process, the problem and screenshots of the errors you are getting.

We had some problems with our game when using a Mac computer. On Windows or Linux the game flows really well, yet on a Mac computer when choosing two cards the speed is very fast. Overall, the game flows and works properly.

- Did you start implementing part of the project? Describe the process and screenshots of your demo.

Our project has a really good base to it so far. The screenshots above ^ show that we have the single player game running well thanks to Kevin expertise in coding. Kevin worked on the Single player portion of our program. Somali designed the options page. I, Nicole worked on the Multiplayer portion of our project. Ibrahim worked on the Statistics page.

3. Take screenshot of your github or gitlab branches showing commits that all the team members have made so far.



Kevin has created most of our base so our cardDeck class, main menu. Somali has helped code classes and create designs for buttons. I have made the Kanban board and have uploaded to management files to help keep us organized and code on the side. Ibrahim has helped fix code and still understanding his portion of code.

4. Share your thought about your project management experience. This can vary depending of how many team members are actively participating in the project and the team size, so please describe how is your team dynamics. Some questions that you can answer include:

- Have you changed the scope of the project? Did you define your project rubric based on the template provided here (Links to an external site.)Links to an external site.?

Our project scope is still the same. We all still have the same end goal and our working together in order to achieve that goal. Again, being open and communicating was one of our criteria's for completing our project and we have stuck with that.

- Have you been following the Gantt Chart to control the time invested in the project and engage all the team member? Do you find useful the use of the Gantt Chart?

We have and have not been following the chart. Scheduling can be difficult especially with a group of people who work and have other priorities (classes). Our Gantt Chart is helpful in a way to keep track of who is working on what and meeting deadlines.

- Have you been using a Kanban board to coordinate the activities? Do you find it useful?

Our group has been using the Kanban board. The lists on our board consist of what needs to be done (to-do), what we are currently working on (in progress), and what is

already completed (done). We find Kanban more useful because it is easy most efficient to use.

- Do you have a team leader or collaborative leadership? Is every body participating

Our group is more of a collaborative leadership. Kevin has been coding most of our program for he is bit more familiar with using Github. Everyone has been helpful with any questions or concerns on coding and our project process.

- Are the weekly team reviews useful for your team? Do they help you to set work standards and team communication?

We believe the weekly team reviews have helped. Although our team is very open about any concerns or expressing helpful tips, it is a way to communicate to professor how our progress is going anonymously.

CRC Cards

Class	
Button	
Responsibilities	**Collaborations**
Image Rect Draw button isClicked: handle button clicks	Pygame.Surface Pygame.Rect Pygame.font

Class	**Superclass**
ImageButton	Button
Responsibilities	**Collaborations**
Image Rect	Pygame.Image Pygame.Surface Pygame.Rect

Class	
Card	
Responsibilities	**Collaborations**
Status CardImageDirectories CardImages Position Surface Rect Load card images Switch card status IsClicked: handle button clicks Compare front images Show card image	Pygame.Image Pygame.Surface Pygame.Rect CardStatus

Class	
CardDeck	
Responsibilities	**Collaborations**
Deck CardCount Load cards (10, 14, 18) Check deck status (single and multiplayer) Check all cards face up	Card Pygame.time.wait

Class	
CardStatus	

Responsibilities	**Collaborations**
back enum front enum solved enum	Card

Class	
InputBox	
Responsibilities	**Collaborations**
Rect Color Text Text Surface Active Handle events Update text box when long word occurs Draw input box	Pygame.Surface Pygame.Rect Pygame.font Pygame.color

Class	
Score	
Responsibilities	**Collaborations**
Score Multiplier Raise multiplier Reset multiplier Raise score Display multiplier Display score	TextPrinter

Class	**Superclass**
MultiplayerScore	Score
Responsibilities	**Collaborations**
Score for second player Multiplier for second player Raise second multiplier Reset second multiplier Raise second score Display multipliers Display scores	TextPrinter

Class	
Statistics	
Responsibilities	**Collaborations**
CardsClicked CardsMatched	TextPrinter

GamesPlayed Raise cards clicked Raise cards matched Raise games played Display statistics	

Class	
TextPrinter	
Responsibilities	**Collaborations**
Text Objects Display title Display text	Pygame.font

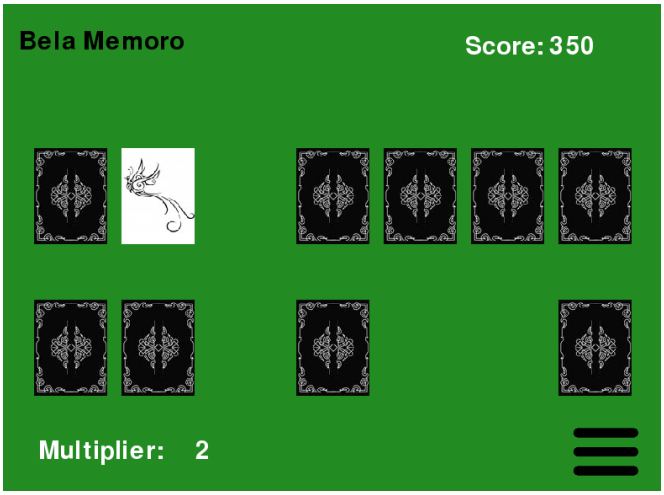
Screenshots (Kevin Zhang)

Operating System: Windows 10

Processor: Intel(R) Core(TM) i3-4030U CPU @ 1.90 GHz

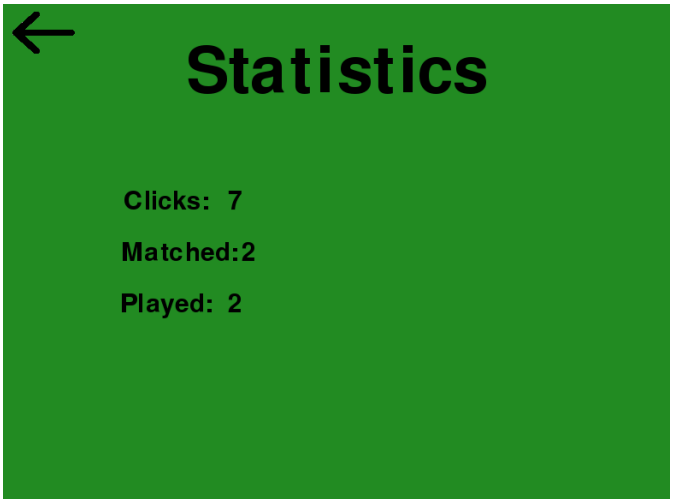
RAM: 6.00 GB

MAC address: 00-1E-64-ED-77-B8


Bela Memoro
Score: 350
Multiplier: 2

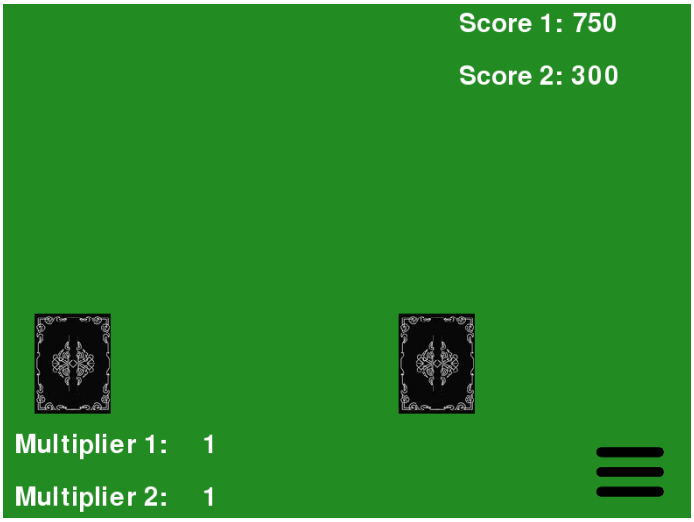

Bela Memoro
Options
How to Play
Music
Credits
X

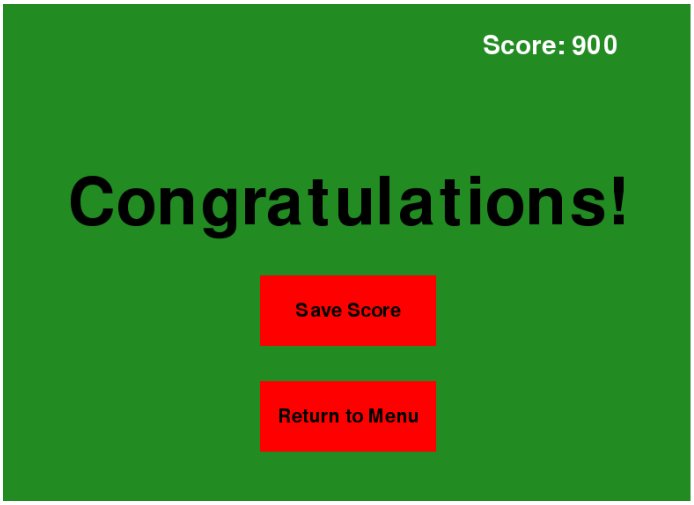

Bela Memoro
How to Play
Welcome to the Python Memory Game!
In Memory, several cards are displayed face down on a board. Each card has a matching pair. The goal of a single player game is to match pairs until all cards on the board are matched. Players will flip up two cards at a time. If the cards do not match, they will flip back over. Be sure to remember their positions! If the cards do match, then points are added to the score, the cards will disappear, and the score multiplier is increased. The score multiplier helps add bonus points onto your score and will increase for every pair you match in a row. However, if you lose your matching streak, the multiplier will reset. Try to get the highest score possible! After a single player game, you can save your

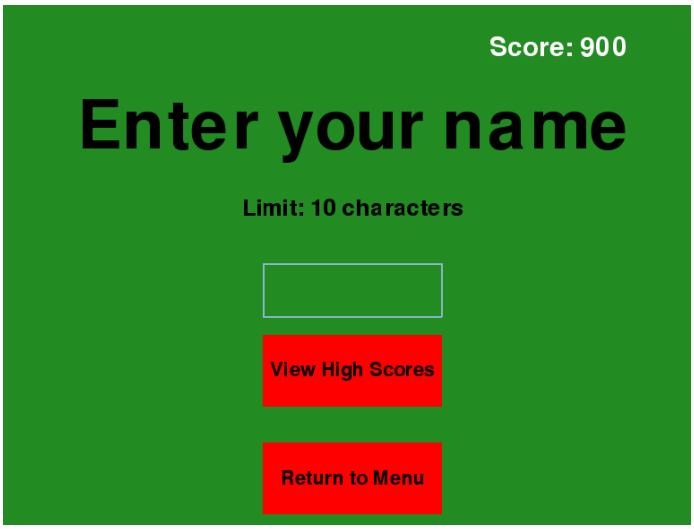

Statistics
Clicks: 7
Matched:2
Played: 2


Bela Memoro
Credits Page
Created by:
Somali Bayi
Ibrahim Hakim
Nicole Racovites
Kevin Zhang
Music: "Netherplace" (Royalty Free Music)
Updated: April 18, 2019


Bela Memoro
Multi Player
Easy (10 cards)
Medium (14 cards)
Hard (18 cards)


Score 1: 750
Score 2: 300
Multiplier 1: 1
Multiplier 2: 1


Score: 900
Congratulations!
Save Score
Return to Menu


Score: 900
Enter your name
Limit: 10 characters
View High Scores
Return to Menu


High Scores (10 Cards)
ke 1000
kevin 850
nicoel 800
asdf 299
qwer 100
test 80
cms 0
fffffffffffff 0
nraco 0
hello 0
Return to Menu
Test Recursion