Course Project Documentation

CS101 Project

TRON++

GROUP NO: 332

.

- Yedukrishnan P: 140010055
- Raunak Shrivastava: 14D170025
- Anish Puthuraya: 140110035
- Ankush Nagulwar: 14D170016

TABLE OF CONTENTS

INTRODUCTION

The games Tron and Snakes are classic and has been a favorites for many of the people for a long time. Although with time, just like most games, Tron started to seem monotonous and slowly lost its fun.

With this as our motivation, we have tried to create three different modes for the classic game and each unique in its own respect hopefully making it a lot more fun to play.

PROBLEM STATEMENT

The aim of the project is to create an independent and totally self-contained game consisting of three modes of the classic Tron game and hence the name Tron++.

The game aims at improving the spontaneous thinking process of the users, while at the same time, being a lot of fun. The idea is to create a user friendly interface for the game and making the three modes - Tron, Curve, and Snake mode unique and fun to play.

REQUIREMENTS

SOFTWARE REQUIREMENTS:

1) Allegro 4.2.3 : This was the graphic library used as it wouldn't have been possible to have the same level of designing solely with the simplecpp package

2) CodeBlocks : The compiler

HARDWARE REQUIREMENTS:

Nothing Specific Required

IMPLEMENTATION

The different modes of the games are:

- **Tron mode**: Player controls a circular head which leaves a trail as it moves. First one to collide with the walls or with the trail of the opponent loses.

- **Snake mode**: write details for the snake mode

- **Curve mode**: write details for curve mode

FUNCTIONALITY:

The main functional areas are:

- The collision function should extract the rgb values from the pixel and check that with the corresponding color. If the color matches, true value is returned.

- In Tron mode, any collisions, either with wall or trail should immediately result in the game ending. Under no other condition should the game end or crash.

- In Snake mode, appearance of arcane fruits should be random and consuming a fruit should result in the appearance of next fruit. The skin shedding should be spontaneous without any lag and collision with oneself is allowed.

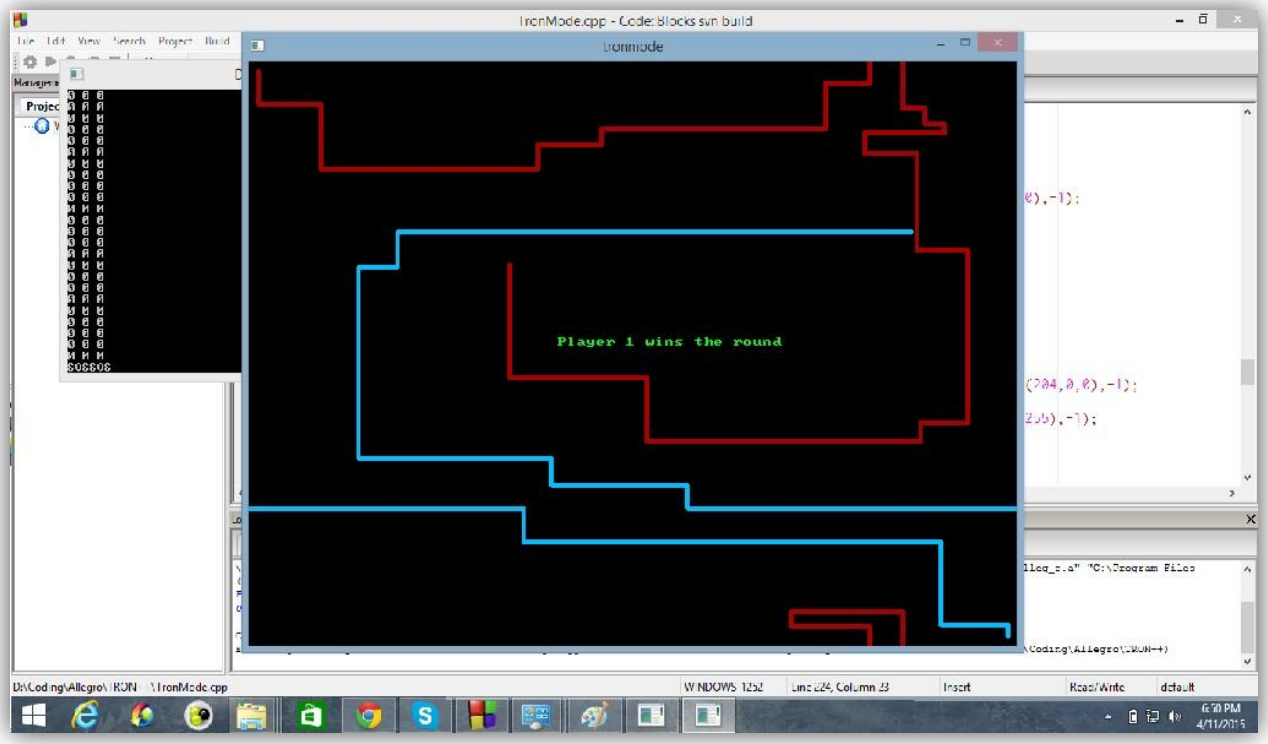
- Player should be able to select from among.
    - Mode: Tron, Snake, Curve.

- There should be an efficient way of tracking the wins by the respective players so that final winner can be selected once the total matches are over.

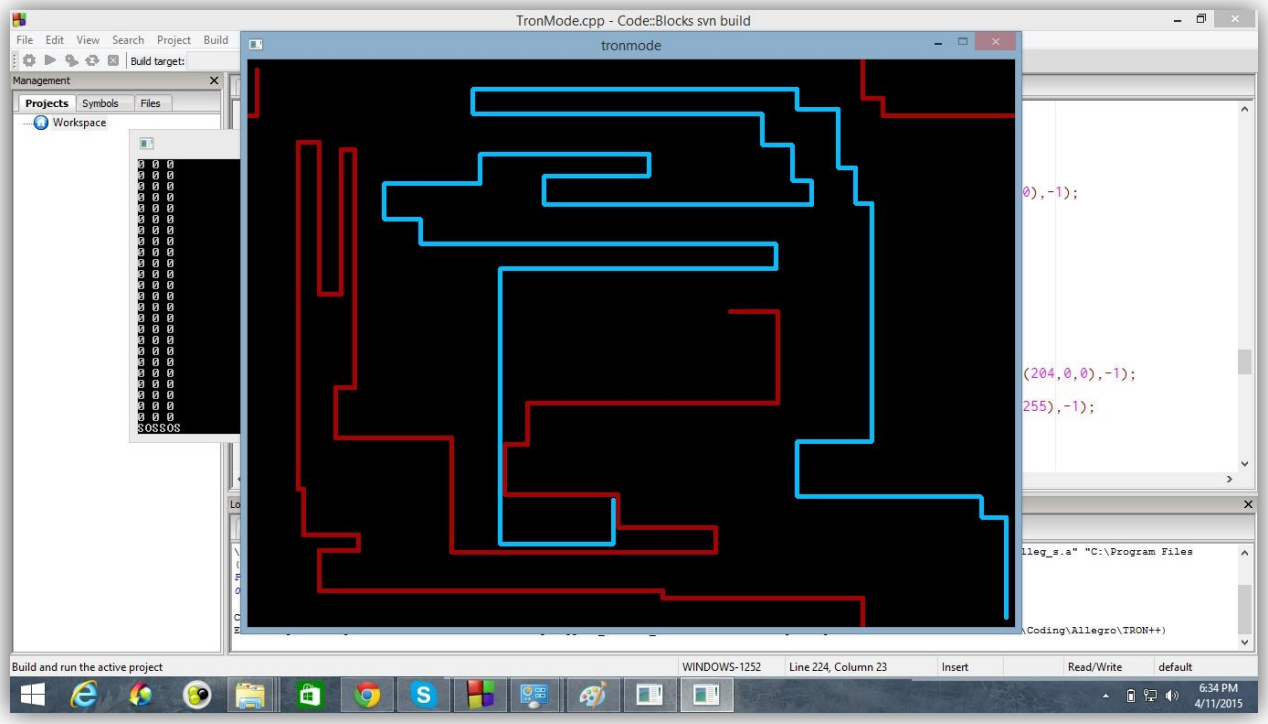
- The close button function should work without much time lag.

TESTING STRATEGY AND DATA

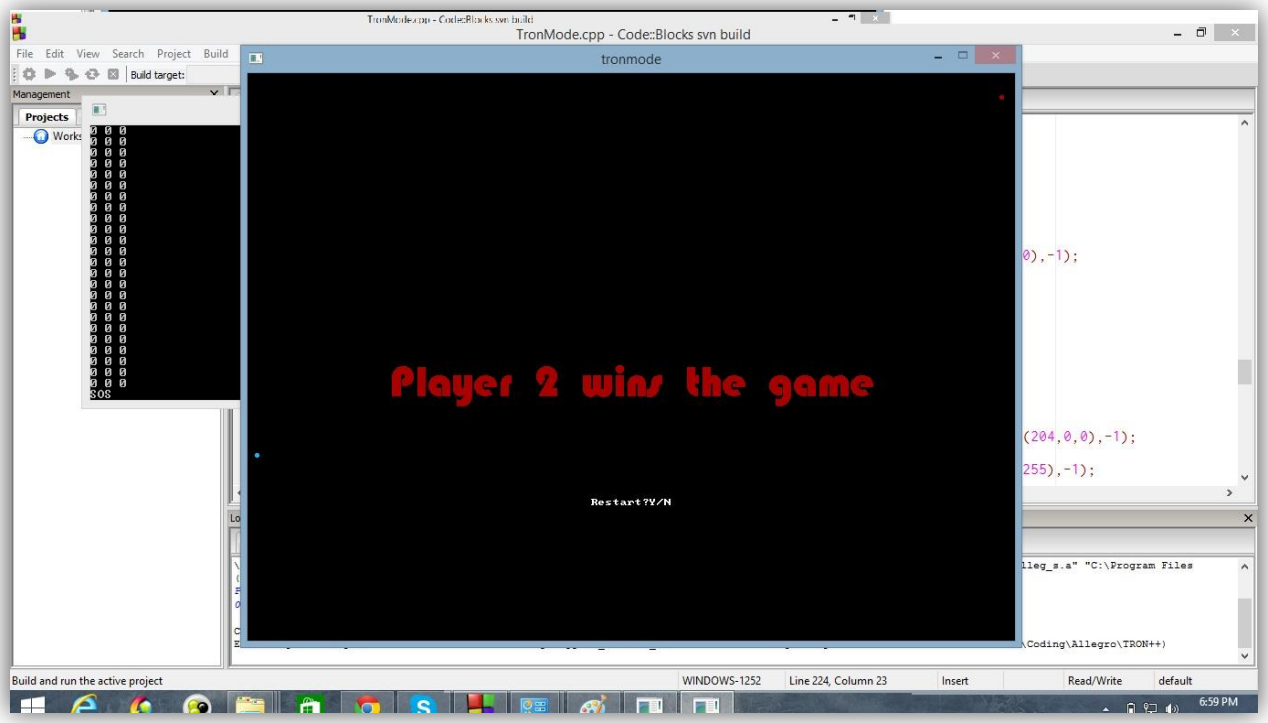
Debugging can be done in the code itself for final testing.

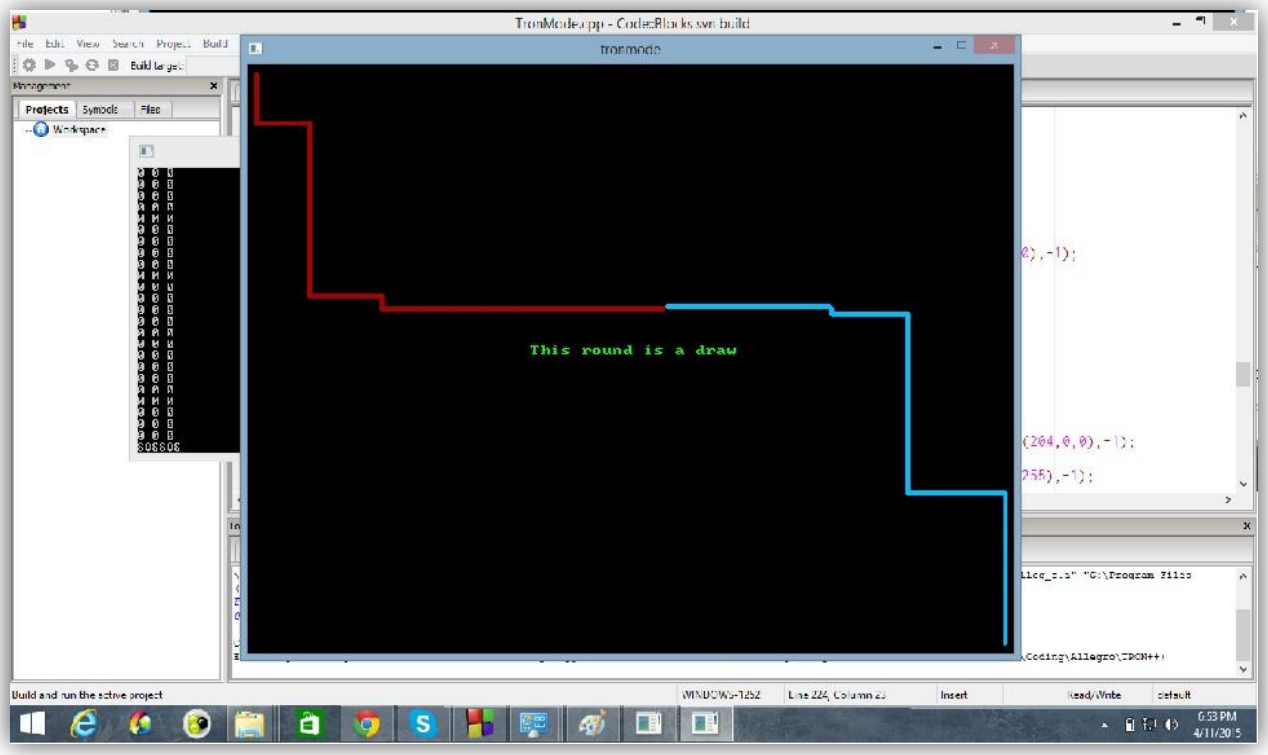


TRON MODE – Player 1 wins the round

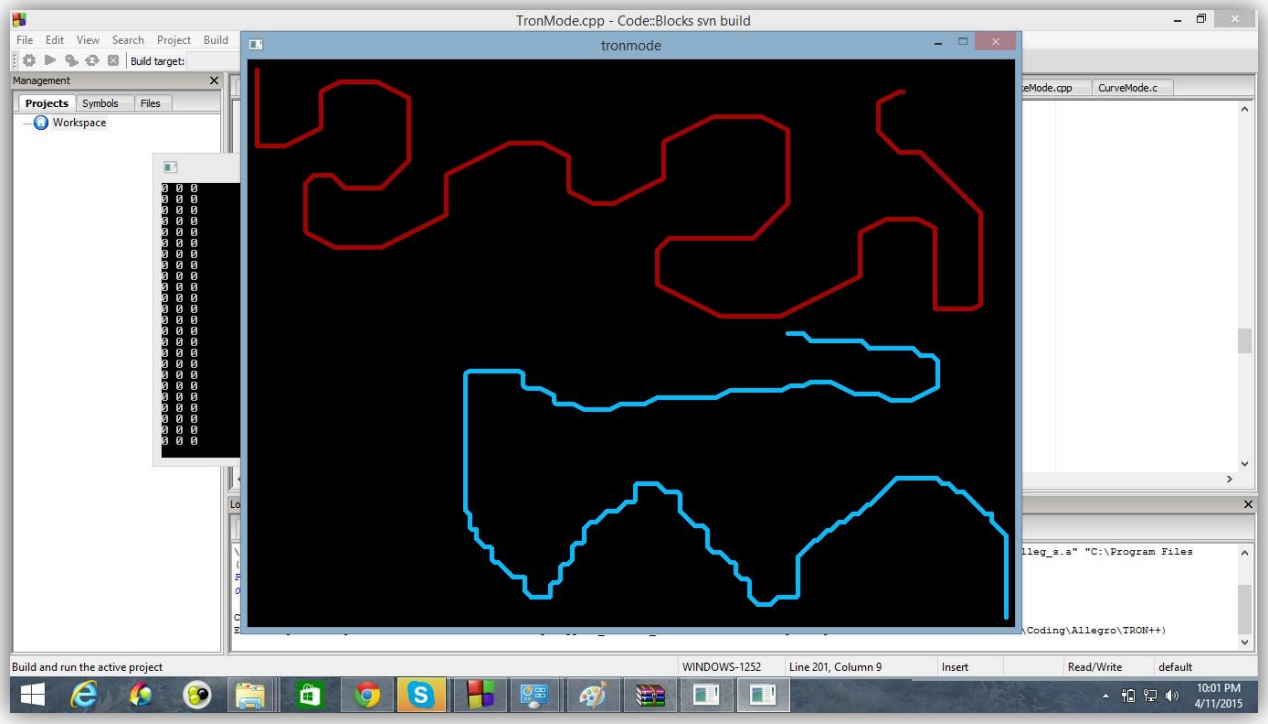


TRON MODE - Player 2 wins the round

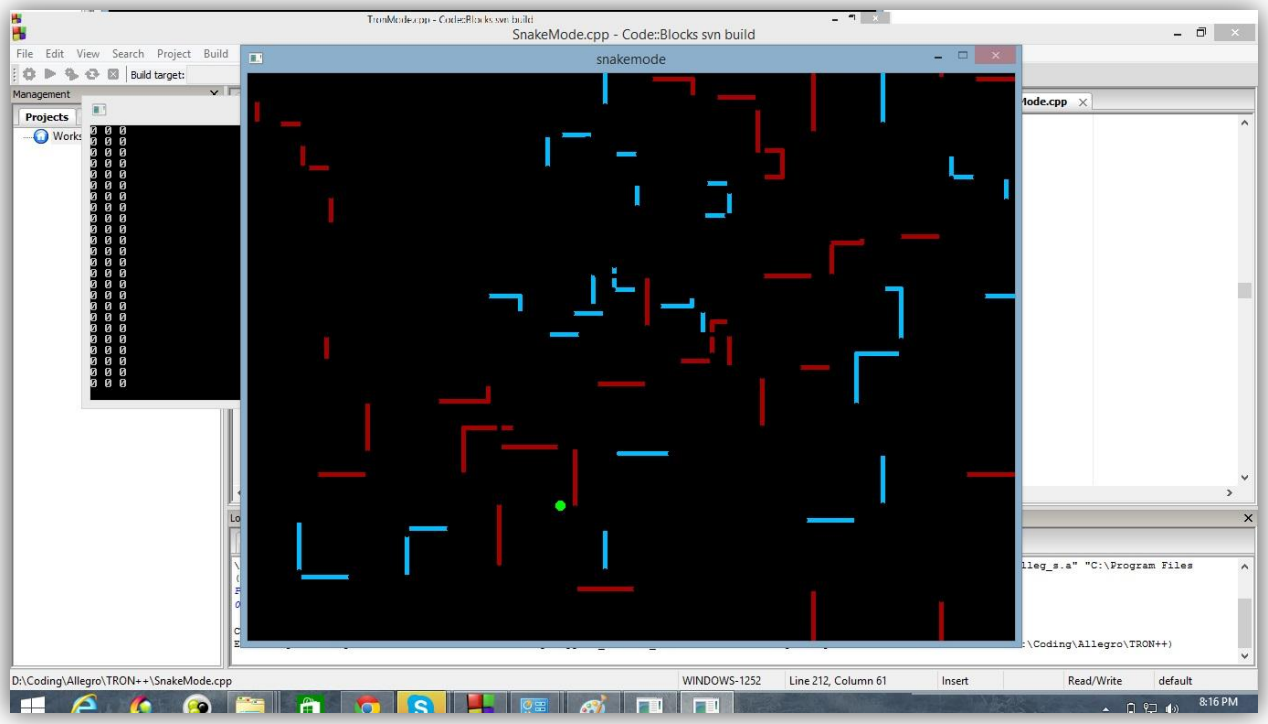




TRON MODE – The round is a draw



CURVE MODE – Screenshot of gameplay



SNAKE MODE – Screenshot of gameplay

DISCUSSING OF SYSTEM

- **What worked out as per the plan?**
  The collision system, based on pixel color segmentation, worked effectively and gave the intended and expected results.

- **What we added more than discussed in the SRS?**
  We improved upon the existing Snake mode and implemented power-ups in the mode to improve gameplay experience.

  The power-ups result in length change in Snake mode and thickness change in the Tron mode, thus making the game more challenging and fun.

- **What changes were made?**
  We didn't make any significant changes from the initial plan we had.

FUTURE WORK

- Introducing the single player and multiplayer option by AI Management with algorithms for controlling the movement of AI.
- Setting up different levels of AI
- Creating more advanced power ups or packages having a variety of effects on the players
- Creating various other obstacle courses.
- Making 3-D version of the same.

CONCLUSIONS

We have created an improved version of the classic game Tron by creating up different modes and making them unique and fun. This game can be used for recreational purposes. At the same time, the code can be improved upon by future users to add more features to the game. At the end of the day, the project was a great learning experience for us and helped us in understanding the core of the course.

REFERENCES

Introduction to C++ by Abhiram Ranade

http://aichallenge.org/.

http://stackoverflow.com/

Allegro manual for everything related to allegro from installing it to basic working

"https://www.allegro.cc/manual/4/"

Allegro tutorials to become more comfortable with allegro

"https://www.youtube.com/watch?v=IZ2krJ8Ls2A&list=PL6B459AAE1642C8B4"