Assignment 5: Separation Axis Test

Course: IGME 309 – Real Time Simulations for Games II

Golisano College of Computing and Information Sciences

School of Interactive Games and Media

Rochester Institute of Technology

Due: Check in MyCourses

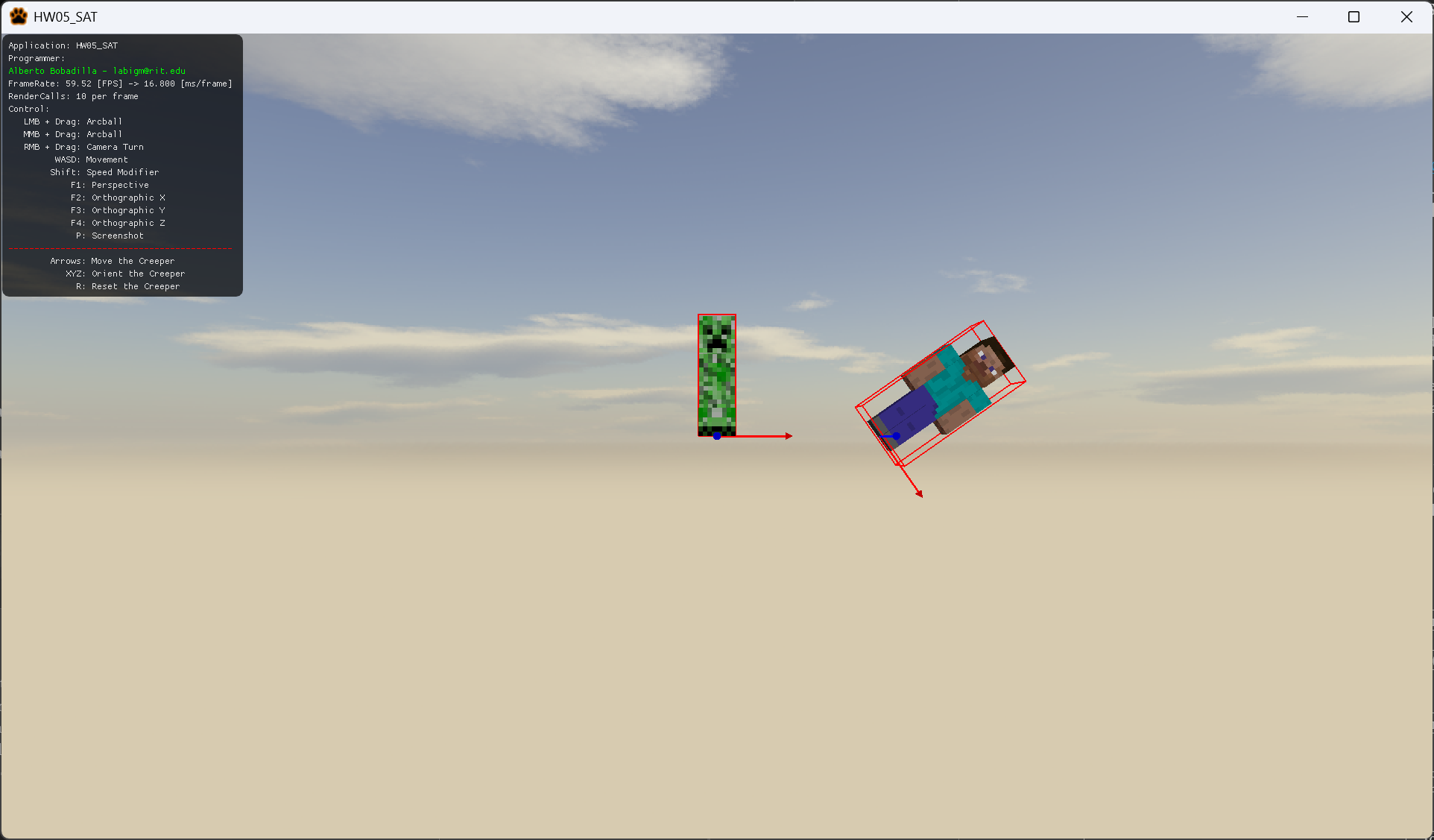
Deliverable: Unzipped MyRiggidBody.cpp file

## Getting Started

Video description: <https://www.youtube.com/watch?v=oDsgx-uM1s0>

In the class repository, I’ve included a solution under the \_Binary folder.

The starter code will give you this out of the box:



## Objective

The objective of implementing the Separation Axis Theorem (SAT) in collision detection is to provide an efficient and reliable method for determining whether two convex shapes are intersecting. By projecting the shapes onto a series of axes (which are typically the normals to the edges of the shapes), SAT checks if there exists an axis along which the projections of the two shapes do not overlap. If such an axis exists, the shapes are not colliding; otherwise, they are. This method is particularly useful in real-time applications such as games and simulations, as it allows for fast and accurate collision detection between complex objects with minimal computational overhead.

## Grading Breakdown

|  |  |
| --- | --- |
| Criteria | Percentage |
| Collision Detection using SAT |  |
|  | 100% |

## Deductions:

|  |  |
| --- | --- |
| Issue | Penalty |
| Hardcoding any values (e.g., model dimensions, axis calculations, etc.) |  |
|  | -20% |
| Failure to comment code adequately for clarity and understanding |  |
|  | -20% |

## Extra Credit:

|  |  |
| --- | --- |
| Criteria | Percentage |
| 15 Separation Planes Implementation: Successfully generate the 15 separation planes as required by the SAT algorithm. |  |
|  | +20% |

## Additional Instructions

Ensure you adhere to the coding standards discussed in class, including proper indentation and clear variable naming.  
Submit your completed assignment through the dropbox labeled A5 – Separation Axis Test by the due date.

You have the option to approach this assignment by either coding it from scratch, manually calculating and applying all necessary mathematical operations, or by utilizing the "Real-Time Collision Detection" book as a reference. Alternatively, you may leverage an AI agent to generate pseudocode, which must then be integrated into our framework or an equivalent solution. Regardless of the approach you choose, please ensure that you include a comment at the beginning of your SAT method indicating the specific approach you employed to solve this problem.