

Tunku Abdul Rahman University College

**Jaegers Prototype**

**BACS 2173**

Graphics Programming

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Student Names** | **Student ID** | **Contribution (%)** | **Sign** | **Grade (for Tutor)** |
| 1. Lim Kah Yee | 20WMR09487 | 50 |  |  |
| 2. Joan Hau | 20WMR09465 | 50 |  |  |
| **Total** | | **100%** |  |  |

|  |  |  |
| --- | --- | --- |
| Programme | : | RSF2 |
| Tutorial Group | : | G5 |
| Date Submitted | : | 6 September 2020 |

# Introduction

The Jaeger that we created for the upcoming Pacific Rim Uprising sequel movie is named RGM-79C. The RGM-79C Gundam is a fictional manned robot and was the first mobile suit mass produced by the Earth Federation. The reason we choose this model as our giant robot for the upcoming sequel is that the exterior appearance of the jaeger is a classical design and able to attract the attention of the audience compared to the mecha that appeared in the previous Pacific Rim Uprising which is more fashionable. Besides, the design of the jaeger represents the symbol of the feminine as we thought that it could gain and attract the attention of the female audience as well. The Jaeger that we designed is based on the concept of the Gundam that appeared in the recent novel <Mobile Suit Gundam: The Blazing Shadow> which was written by Toshiyuki Itakura  in 2013.



# System Specifications

## Integrated Development Environment (IDE)

The Integrated Development Environment(IDE) we choose for our assignment for creating and designing the robot (RGM-79C) is Microsoft Visual Studio 2017. The reason that we choose to use this IDE for creating and designing the robot is because Microsoft Visual Studio is an IDE that is developed by Microsoft Corporation, it allows us to easily develop the robot (RGM-79C) in Microsoft platform. Besides, Microsoft Visual Studio is fully compatible with the open source graphic library which is OpenGL that we will be used to creating and designing the robot (RGM-79C). Therefore, Microsoft Visual Studio 2017 is our first choice IDE to be used.

## Programming Language

The programming language that we choose to design and develop this robot (RGM-79C) is C++ language since this language provides few benefits for us when we develop the robot (RGM-79C).

Firstly, C++ can **reduce the latency** of our robot’s Frames Per Second(FPS) when it is showing in the Windows Form. For example, by using C++, an optimized application can run even on low-end devices that do not have any high computation power available. Therefore, C++ able renders our robot faster and prevents the robot from becoming choppy and laggy due to low FPS.

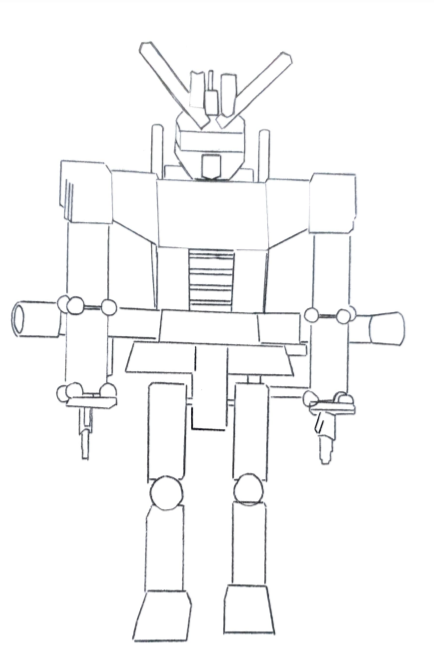
Secondly, C++ able provides us the **provision of total control over memory management** when we develop the robot. For example, we will be able to observe the memory usage of  our robot when it is running in the Windows Form. This is very important since high memory allocation for our project may cause the computer overload and crash of the Windows Form. Therefore, by using C++, we can observe and optimize the memory allocated for the robot to prevent the computer overload.

## Graphics Library Used

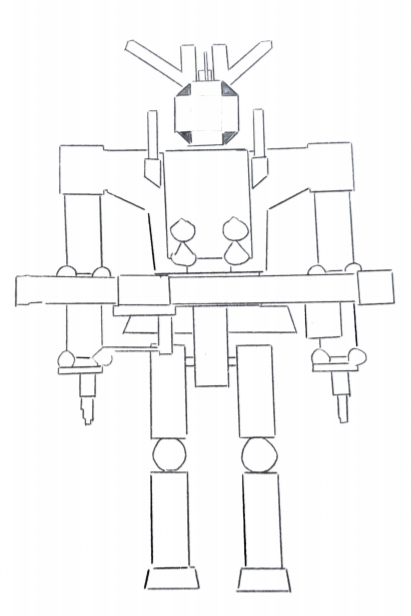
The graphics library that we use in this project is Open Graphics Library (OpenGL) that was initially released by Silicon Graphics, Inc. OpenGL is a cross-language, cross-platform Application Programming Interface(API) for rendering 2D and 3D vector graphics in application programs. OpenGL provides an API for the application program to interact with the Graphics Processing Unit(GPU) to achieve hardware-accelerated rendering.

# Design Concept

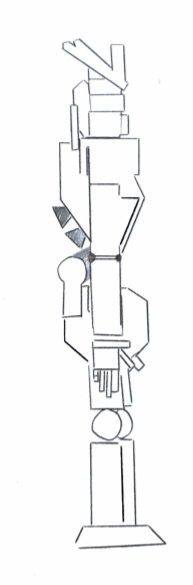
## Front View



## Back View



## Side View



# Primitives Used

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | QUADS | SPHERE | TRIANGLE | POLYGON | CYLINDER | TRIANGLE\_FAN | LINE | LINE LOOP |
| **HEAD** | 76 | 0 | 10 | 2 | 1 | 0 | 0 | 0 |
| **LEFT ARM** | 139 | 8 | 4 | 0 | 0 | 0 | 0 | 0 |
| **RIGHT ARM** | 139 | 8 | 4 | 0 | 0 | 0 | 0 | 0 |
| **BODY** | 42 | 0 | 0 | 0 | 0 | 0 | 4 | 0 |
| **LEG** | 66 | 4 | 0 | 0 | 0 | 0 | 2 | 0 |
| **BACKPACK** | 6 | 0 | 0 | 0 | 4 | 0 | 0 | 0 |
| **WEAPON** | 102 | 1 | 0 | 0 | 13 | 22 | 0 | 11 |
| **SHIELD** | 43 | 0 | 0 | 2 | 0 | 0 | 0 | 30 |
| **TOTAL** | **613** | **21** | **18** | **4** | **18** | **22** | **6** | **41** |
| **GRAND TOTAL** | **743** | | | | | | | |

# User Manual

## Whole Robot

|  |  |
| --- | --- |
| **Key** | **Action** |
| ‘W’ | Walking Animation (front, left, right) |
| ‘S’ | Stand |
| VK\_LEFT | To Rotate Robot to Left |
| VK\_RIGHT | To Rotate Robot to Right |
| VK\_TAB | To Stop the Action of the Robot |
| VK\_SPACE | To Reset All Settings |

## Robot Head

|  |  |
| --- | --- |
| **Key** | **Action** |
| ‘N’ | To Move the Robot Head to Left and Back to Origin |
| ‘M’ | To Move the Robot Head to Right and Back to Origin |

## Robot Arm

|  |  |
| --- | --- |
| **Key** | **Action** |
| ‘H’ | To Perform Both (Left && Right) Arm Movement |
| ‘K’ | To Move Left Arm Up and Down |
| ‘P’ | To Move Right Arm Up and Down |
| ‘V’ | To Move Left Lower Arm Up and Down |
| ‘Z’ | To Move Right Lower Arm Up and Down |

## Robot Finger

|  |  |
| --- | --- |
| **Key** | **Action** |
| ‘F’ | To Activate/Deactivate Finger |
| VK\_UP | To Close the Finger |
| VK\_DOWN | To Open the Finger |

## Robot Leg

|  |  |
| --- | --- |
| **Key** | **Action** |
| VK\_SHIFT | Move Left Leg Up |
| VK\_CTRL | Move Left Leg Down |
| VK\_UP | Move Right Leg Up |
| VK\_DOWN | Move Right Leg Down |

## Weapon & Shield

|  |  |
| --- | --- |
| **Key** | **Action** |
| VK\_F1 | To Activate/Deactivate Weapon and Shield |
| ‘G’ | To Do Preparation for Fire |
| ‘F’ | First Time - To Accumulate Beam  Second Time - To Fire Beam |

## Shield

|  |  |
| --- | --- |
| **Key** | **Action** |
| VK\_F8 | To Activate Attack Mode (Shield Shown) |
| ‘B’ | To Move Shield Up and Down (in Attack Mode) |

## London Bridge

|  |  |
| --- | --- |
| **Key** | **Action** |
| VK\_F9 | To Activate/Deactivate London Bridge |
| ‘W’ | To Move Middle Bridge Up |
| ‘S’ | To Move Middle Bridge Down |

## Projection

|  |  |
| --- | --- |
| **Key** | **Action** |
| VK\_F2 | To Change View between glOrtho or gluPerspective |
| ‘2’  / VK\_NUMPAD2 | To Zoom Out the View (gluPerspective view only) |
| ‘8’  / VK\_NUMPAD8 | To Zoom In the View  (gluPerspective view only) |
| ‘4’  / VK\_NUMPAD4 | To Move View to Left (both view) |
| ‘6’  / VK\_NUMPAD6 | To Move View to Right (both view) |
| ‘5’  / VK\_NUMPAD5 | To Move View Up (both view) |
| ‘0’  / VK\_NUMPAD0 | To Move View Down (both view) |
| ‘7’  / VK\_NUMPAD7 | To Rotate View to Left (glOrtho view only) |
| ‘9’  / VK\_NUMPAD9 | To Rotate View to Right (glOrtho view only) |

## Texture Changing

|  |  |
| --- | --- |
| **Key** | **Action** |
| VK\_F5 | To Activate/Deactivate Change Robot Texture |
| Number Key 1 - 9 | To Change Robot’s Texture |
| VK\_F6 | To Activate/Deactivate Change Background Texture |
| Number Key 1 - 6 | To Change Background’s Texture |

## Lighting

|  |  |
| --- | --- |
| **Key** | **Action** |
| VK\_F7 | To Control Light On / Off |
| ‘8’  / VK\_NUMPAD8 | To Move Light Up |
| ‘2’  / VK\_NUMPAD2 | To Move Light Down |
| ‘4’  / VK\_NUMPAD4 | To Move Light Left |
| ‘6’  / VK\_NUMPAD6 | To Move Light Right |
| ‘7’  / VK\_NUMPAD7 | To Move Light Near |
| ‘9’  / VK\_NUMPAD9 | To Move Light Far |