**Week 0 (21 Jan 12):**

* Did 2D XNA game tutorial (gunbound like game without movement)

**Week 1 (23 Jan 12):**

* Did tutorials for sprite animating, additive blending and particle engine
* Revised C#
* Found and aligned Byakuya sprites in photoshop
* Created Byakuya test class in blue background
  + Move list: idle, running (4 directions), jumping, dashing (4 directions), ducking.

**Week 2 (30 Jan 12):**

* Analysed Platformer tutorial for collision handling and level framework
* Started BleachGame project.
* Created tile-based level similar to Platformer tutorial.
* Created new byakuya character class with new algorithm:
* Animation player instead of enumumeration states.
* Gravity implemented
* Move list: idle, running (2 directions), jumping, dashing (4 directions), ducking.
* Implemented collision handling:
* customised bounding box for individual animations.
* Dash collision handling to prevent forward and backward tunneling. (Backward tunneling occurs when we tunnel backwards when back facing a wall due to the dash sprite being larger than the other sprites).

**Week 3 (6 Feb 12):**

* Implemented side-scrolling level with minimal understanding of code.
* Restructured character class into an abstract base class, inherited by Byakuya class.
* Learn Photoshop techniques for sword trail.
* Drew simple sword trail using Gaussian noise filter / crystallization filter / perspective transformation / shear filter /motion blur filter. Sword trail was implemented directly as part of the sprite.
* Added to move list: groundSlash.

**Week 4 (13 Feb 12):**

* Restructured character and Byakuya class for better code decoupling. Use character state to determine character actions like a state machine – easier to restrict actions in states.
* 3 main functions are used:
* GetInput: Solely for mapping input keys to attempted actions.
* UpdateCharacterState: Get current character state based on previous character state and attempted actions. Handles flipping of sprites and action priorities.
* ApplyCharacterActions: Apply the movements, and changes state if necessary for timed animations.
* Implemented 2nd level ground Slash.
* Implemented Jump and slash – able to rise if jump is still pressed while air-slashing.
* Implemented particle engine for shikai.
* Added to move list: airSlash, Shikai.
* Tested Reiatsu drawing for shikai.

**Week 5 (20 Feb 12):**

* Created sprites for shikai mode without sword and with pink reiatsu.
* Implemented shikai mode. Automatically exit shikai mode after some time. Implemented animation for returning to normal mode.
* Implemented 2 levels of shikai ground attack. Use 2 different types (particles and textures) for senbonzakura particles.
* Implemented shikai air attack. Need more fine tuning for senbonzakura air path.

**Week 6 (27 Feb 12):**

* Separated senbonzakura paths from attack function. Fine tuned senbonzakura paths.