JOUST

Published by Williams Electronics, released for arcades in 1982. Also published by Atari, Inc. for several Atari versions, Apple II, and Max. HAL Laboratory also ported Joust to the NES.

* <https://www.free80sarcade.com/nes_Joust.php>
* <https://en.wikipedia.org/wiki/Joust_(video_game)>

< Description>

**Planned Scope**

My plan is to create a clone of Joust with the following:

* Flying mechanic – flap to fly
  + Risk/Mitigation: just flying around screen (no platforms/gravity)
  + Goal: slow flapping
  + Stretch: slow + fast flapping (2 speeds)
* The original 3 enemy jousters
  + Bounder
  + Hunter
  + Shadow Lord
  + Stretch: pterodactyl
* “Bouncing” when collision with altitude tie (no kill)
* Player lives – start with 3
* Waves – level progression (x number of specific enemies each wave)
  + Risk/Mitigation: 1 wave
  + Goal: > 3 waves
  + Stretch: endless repeatable waves after a certain point
* A basic Main Menu and game start/end loop
* Sound effects

**Omissions / Stretch Goals**  
(note that some of these may be repeated from above)

* Player immunity on respawn
* Wall wrapping (left/right, NOT top/bottom)
* Egg drop / enemy spawn mechanic
* Multiplayer (up to 2 players)
* Wave based progression (potentially up to endless mode)
* Different backgrounds for different waves
* Pterodactyl enemy type
* Background music during game
* High Score display and tracking (store as a file perhaps?)

**Implementation**

NOTE: Time completions will be written as (hrs:mins)

MS1: Character moving in 4 directions (no flapping)

* Time Estimation: 16 hours
* Completion date estimation: 11/16
* Details: Initial project setup, asset gathering, initial character (no animations), input system. Character will be able to move left/right/up/down at first
* Actual time: 4:50
* Actual completion date: 11/16

MS2: Add the first enemy & collision system

* Time Estimation: 8 hours
* Completion date estimation: 11/18
* Details: Add enemy with basic enemy system that is extensible. Setup collision detection and reaction system.
  + Note for killing: killing entity’s center must be above the other enemy’s center and lance must be colliding. Killing an enemy should give the player points as well.
* Actual time: 13:55
* Actual completion date: 11/18

MS3: Flying/Gravity system on player/enemies

* Time Estimation: 12 hours
* Completion date estimation: 11/21
* Details: Add gravity to players and enemies. Give player the ability to flap wings (no animation yet) to fly.
* Actual time: 4:57
* Actual completion date: 11/19

MS4: Enemy AI

* Time Estimation: 20 hours
* Completion date estimation: 11/23
* Details: Give enemy “random” movement and flying to move around screen/map. Move out common functionality of player & enemies to the entity class so both can call said functionality.
* Actual time:
* Actual completion date:
* Notes: This shippable is not yet complete, and will have continuous work throughout project. Initial “Bounder” AI is setup to be pretty stupid. Currently all enemies are using the same AI but with different speeds. Plan is to continue game development then go back and tweak enemy AI near end of project.

MS5: Layout the map and setup waves

* Time Estimation: 10 hours
* Completion date estimation: 11/25
* Details: Layout map with sprites that have colliders to act as walls/platforms. Set player respawn locations. Setup a system to create and run waves with enemy spawning. Should allow specific enemies to be added along with their spawn locations.
* Actual time:
* Actual completion date:

MS6: Give player lives

* Time Estimation: 4 hours
* Completion date estimation: 11/26
* Details: Give player a set number of lives at the beginning of game. Add player respawning after death. End game when player is out of lives.
* Actual time:
* Actual completion date:

MS7: UI Pass

* Time Estimation: 16 hours
* Completion date estimation: 11/29
* Details: Setup and add UI to game loop, such as score and lives display. Anything not changing at all will be part of the background image itself. Add Main Menu (image w/ “press <button> to start” and a quit option). End game screen should show user their score and allow them to loop back to main menu.
* Actual time:
* Actual completion date:

MS8+: Polish / Stretch Goals (NOTE: animations should be moved upwards, probably between MS3 – MS4)

* Time Estimation: Unknown / As much as possible with remaining time
* Completion date estimation: 12/2
* Details: This includes animations as well as anything in the “Planned Scope” section that has been marked as ‘Stretch’ in addition to anything in the “Omissions / Stretch Goals” section. The order of these goals will be decided upon reaching or getting close to this shippable assuming there is time to complete any of them.
* Actual time:
* Actual completion date:

**Asset Plan**

* Character and enemy sprites, platforms, basic UI assets: <https://www.spriters-resource.com/search/?q=Joust>
* Sounds: Not sure where to find those, will be looking for a pack of those somewhere (similar to how I got them for Space Cadet Pinball). If necessary, will record my own sounds to use in place of the originals.
* The background of the game is black, so if I’ll start with that. If time provides, I’ll change the background itself. Currently no plan what images to put here, but that is a stretch goal.
* Music: This is a stretch goal. If I have time, I will look for some similar to how I’ll find sounds (but I won’t make my own music if I can’t find them).

**Risks and Mitigations**

* Time/Scope (72 planned hours)
  + This risk is most certainly the scariest for me. Considering I spent ~98 hours on the Assembly Game and the estimates were much lower, I expect my planned hours on this game to take longer than my current estimates.
  + I setup the schedule to get a basic version of the game with minimal mechanics earlyish on in case other tasks take much longer.
  + Another risk here is that I have personal obligations over the duration of this assignment as one of my sisters is in town as well as celebrating both of my sisters’ birthdays. I gave padding in the schedule during those dates as well set my “completion date” for the shippable game a weekend before the showcase. The extra time can be spent adding polish and stretch goals or bug fixing and finishing the final implementation.
  + For several of the features I planned to add, in the “Planned Scope” section I added a mitigation version in case I’m running out of time to meet the original goal.
* Assets
  + Finding sprites and clipping them is a weakness of mine, specifically clipping them to use in game. To help mitigate this I’ve saved and am using the site: <https://www.spriters-resource.com/search/?q=Joust>
  + While I’ve found player sprite sheets and sprites for enemies and platforms, I may need more for the UI.
  + If I can’t find more assets I need, I’ll play the emulator and use screenshots to gather the remaining ones I need.
  + A mitigation here would be to use basic shapes if I can’t get certain assets, or potentially make my own main menu / quit game screen.
* Knowledge of Game Framework
  + Prior to writing any lines of code, I need to spend time to read through the provided game framework and understand what is available to me and how to use it.
  + This time is NOT included in my Time/Scope estimation.
  + A mitigation to this is to dedicate time every day starting now towards reading through it. Time between assignments, when walking home, watching tv, time prior to bed, etc. Pretty much find small times to read through sections of it to gain a full understanding of what all the code is doing.