Optimization & Debugging
But first, your Endless Runner
Logically comment your source to demonstrate that you understand how each section works. (0.5)

Your file structure is organized logically and legibly (0.5)

Game runs from localhost (0.5) with no code errors (0.5).

Submit your project to Canvas as a .zip that includes the framework so the graders can run it. (0.5)

https://canvas.ucsc.edu/courses/26569/assignments/86776
## Structure and Design I

<table>
<thead>
<tr>
<th>3 States</th>
<th>Have at least three states: a main menu (0.5), a state where you play the actual game (0.5), and a game over state (0.5). You may name these however you like. You may also have more, depending on how you structure your game.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Collision Detection</td>
<td>Properly use collision detection (0.5).</td>
</tr>
<tr>
<td>Instructions</td>
<td>Communicate how to play w/ clear instructions (0.5).</td>
</tr>
<tr>
<td>State Transitions</td>
<td>Properly transition between states and allow the player to restart w/out having to reload the page (0.5).</td>
</tr>
<tr>
<td>Sound Effects</td>
<td>Use sound effects for key mechanics and/or events (0.5) according to your design.</td>
</tr>
<tr>
<td>Player Input</td>
<td>Have some form of player input/control (0.5) according to your design.</td>
</tr>
<tr>
<td>Randomness</td>
<td>Use randomness to generate challenge, e.g. terrain, pickups, etc. (0.5).</td>
</tr>
</tbody>
</table>
# Structure and Design II

<table>
<thead>
<tr>
<th>Animated Character</th>
<th>Include an animated character(s) (0.5) that use a texture atlas (0.5).</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Metric</strong></td>
<td>Include some metric of accomplishment that a player can improve over time, e.g., score (0.5).</td>
</tr>
<tr>
<td>Simulate Scrolling</td>
<td>Simulate scrolling, e.g., tilesprite (0.5).</td>
</tr>
<tr>
<td><strong>Endless</strong></td>
<td>Be theoretically endless (0.5).</td>
</tr>
<tr>
<td>Playable</td>
<td>Be playable for at least 15 seconds for a new player of low to moderate skill (0.5). (DO THE OPPOSITE OF THIS VIDEO: <a href="https://www.youtube.com/watch?v=eb60pnjABGq">https://www.youtube.com/watch?v=eb60pnjABGq</a>)</td>
</tr>
</tbody>
</table>
Creative Tilt (2 points)

Does your game...

<table>
<thead>
<tr>
<th>Technical Interest</th>
<th>...do something technically interesting? Are you particularly proud of a programming technique you implemented? Did you look beyond the class examples and learn how to do something new? (1)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Visual Style</td>
<td>...have a great visual style? Does it use music or art that you created? Are you trying something new or clever with the endless runner form? (1)</td>
</tr>
</tbody>
</table>
Game Development Crimes

https://twitter.com/fullbright/status/107362481107016704

Dear gamers: all games are duct-taped together by the hopes and dreams of developers.

Dear developers: all games are built as shitty as yours is, especially considering that one thing you did that time when you just couldn't figure it out but hey it works and you'll fix it never.

Steve Gaynor @fullbright
please share your most embarrassing game dev crimes. Mine: all the scripting in Gone Home is contained in 2 gigantic uScript graphs, bc I didn't know about interlinking many smaller graphs at the time. It's only 2 bc the first one started running too slow to navigate. Your turn.

Show this thread

3:27 AM - 16 Dec 2018
Definitely don't do these

Jake Rodkin @ja2ke · 15 Dec 2018
Replying to @fullbright
the final scene for the final TTG Sam & Max game can never be changed, because I baked the lights in Maya, exported the lightmaps to the game engine, checked it in, then slammed every program on my PC shut because we were supposed to be locked hours ago. never saved the Maya file

Pat Scott @_Stralor · 14 Dec 2018
Replying to @fullbright
There is only one file in the first game I ever coded: a single class encapsulating everything.

UI draw calls, game logic, text parsing, etc. 1000s of lines.

Bonus: I had no idea what "static" meant, nor how to pass non-static objects. So I made everything static. Everything.
Matthew Brooks @brooksoid · 15 Dec 2018
Repeating to @fullbright
Every game entity in Wolverine's Revenge contains a single bit determining whether it's a helicopter taking off or not. Including Wolverine.

WONTFIX

Glen Watts @MrFlabyo · 15 Dec 2018
Repeating to @fullbright
The dog in fable 2 can't turn on the spot (lacked the animations for it), so sometimes the AI get stuck running in a small circle trying to get to a goal it can't ever reach. We gave up trying to fix it because 'dogs do that'. No one in QA ever filed a bug for it.
Some sims need warm-up time
Lee Perry Yo! @MrLeePerry · 16 Dec 2018
Replying to @fullbright

Omg so many.

I had some garbage "blowing around" in a 64x64 grid on the map. I built piles of meshes attached to tiny invisible squadmates. It worked.

I don’t know if I ever told anyone that, haha.

7 · 40 · 421

Brendan Walker @binarys · 15 Dec 2018
Replying to @fullbright

Not mine, but one I secretly loved. On “Pirates of Calypso,” the designer needed a fire ship. Ships didn’t support NPCs at that time, but had NPC stations. NPCs did support emit... set them on fire D:

3 · 128 · 1.4K

Steven Lumpkin @SilentOsiris · 16 Dec 2018
Replying to @fullbright

Everything cool in The Secret World’s boss fights is a testing thing. A crash test dummy at a spot, giving it the same effect as if you had cast a spell.

(This is very normal, but it sure sounds weird.)

1 · 9 · 102

Robin-Yann Storm @RVStorm · 16 Dec 2018

For World of Warcraft entities this used to be a term for invisible bunnies. Just invisible bunnies all over the place, for all kinds of purposes.

3 · 10 · 81

When you rode a horse, we simply put the horse inside the player, and spawned a pair of pants that looked like your horse, which you then equipped and wore.

When we first did this, however, we forgot to make the horse stop acting like a horse. Pretty soon there was a rash of server crashes because the horse inside the player was wandering around, picking up the stuff it found inside the player, rifling through the player's backpack and eating things it thought were edible, and eventually, wandering "off the map" because the player's internal coordinate system was pretty small, and the edges weren't impassable.
Don't (deliberately) do crimes

But games aren't realistic simulations and sometimes you just hack in a solution and go work on other things instead.
Game Physics are not real physics

https://twitter.com/AIandGames/status/1149999152551604224
Optimization: Finding ways for the computer to do less work
Frames

One second

Thirty frames per second:
~33.33 milliseconds per frame

Sixty frames per second:
~16.67 milliseconds per frame
Frame Budget

One frame

updateLogic  Render  event
collision  particles  sprites
Profiling

Luckily for us, browsers have built-in profiling tools
Profiling

We can drill down and see how many microseconds each operation is taking.

\[ ms = \text{millisecond} = \frac{1}{1000} \text{th} \]

\[ \mu s = \text{microsecond} = \frac{1}{1,000,000} \text{th} \]
"We should forget about small efficiencies, say about 97% of the time: premature optimization is the root of all evil. Yet we should not pass up our opportunities in that critical 3%"
How to do less work

But first check to see if what you are doing even matters

Make the calculations cheaper
Make the inner loop faster
Do fewer calculations
Don't run everything every frame
Debugging
Useful random debugging advice

1. When you find a problem, change something so that same problem can't happen again
   a. assert()
   b. Keep a debugging notebook

2. Make debug tools
   a. Quicker feedback is better
   b. Display values live if possible

3. Only make one change at a time and then test it

4. Just because you paused the game doesn't mean it's paused
   a. And stopping one update doesn't mean you stopped all of them

5. console.log() is slow
   a. Faster to print an array as a string than to individually print the contents
Useful random debugging advice

1. When you find a problem, change something so that same problem can't happen again
   a. assert()
   b. Keep a debugging notebook

2. Make debug tools
   a. Quicker feedback is better
   b. Display values live if possible

3. Only make one change at a time and then test it

4. Just because you paused the game doesn't mean it's paused
   a. And stopping one update doesn't mean you stopped all of them

5. console.log() is slow
   a. Faster to print an array as a string than to individually print the contents
Useful random debugging advice

1. When you find a problem, change something so that same problem can't happen again
   a. assert()
   b. Keep a debugging notebook

2. Make debug tools
   a. Quicker feedback is better
   b. Display values live if possible

3. Only make one change at a time and then test it

4. Just because you paused the game doesn't mean it's paused
   a. And stopping one update doesn't mean you stopped all of them

5. console.log() is slow
   a. Faster to print an array as a string than to individually print the contents
Useful random debugging advice

1. When you find a problem, change something so that same problem can't happen again
   a. assert()
   b. Keep a debugging notebook

2. Make debug tools
   a. Quicker feedback is better
   b. Display values live if possible

3. Only make one change at a time and then test it

4. Just because you paused the game doesn't mean it's paused
   a. And stopping one update doesn't mean you stopped all of them

5. console.log() is slow
   a. Faster to print an array as a string than to individually print the contents
Useful random debugging advice

1. When you find a problem, change something so that same problem can't happen again
   a. assert()
   b. Keep a debugging notebook
2. Make debug tools
   a. Quicker feedback is better
   b. Display values live if possible
3. Only make one change at a time and then test it
4. Just because you paused the game doesn't mean it's paused
   a. And stopping one update doesn't mean you stopped all of them
5. console.log() is slow
   a. Faster to print an array as a string than to individually print the contents
AABB characters and slopes

An example of a real-world physics-and-debugging problem in a game with 2D physics like yours

https://twitter.com/eevee/status/1133248372624613376