

# HELICOPTER RESCUE Final Cycle Presentation

Nick VonDollen, Taylor Cook, Michael Orzel, and Bj Blair III 5/1/17



## **Previous Cycle**

Focused on testing existing features

## This Cycle:

- Adding new features
- Testing new and old features
- Polishing old features



# System/Cycle Intent

## **System Intent:**

Create a fun and competitive helicopter rescue game where the player writes MiniAT programs to rescue victims and avoid obstacles.

## **Cycle Intent:**

Add base features from our previous cycle in order to have a minimally complete game, including victim rescue, collision, and to be able to successfully complete a level.

## Design

Design Doc/ Cycle 2/ William Blair, Taylor Cook, Michael Orzel, & Nicolas VonDollen

### **FEATURES**

#### MVPP Feature #3

Name: Helicopter movement

**Feature**: The programmer can apply 'thrust' to the helicopter in the cardinal directions (up, down, left, and right). The thrust has both separate vertical and horizontal values, with a range of -100 (for backwards) and 100 (for forwards). The helicopter will not move instantaneously or stop instantaneously as well. It will have inertia and momentum adding both realism and an extra challenge for the player.

**Constraints:** The helicopter can move up or down on the screen but no portion of the helicopter can move offscreen. If the helicopter hits an object the player will either lose a life or lose the level. The helicopter hitting a victim results in either a save or kill, depending on velocity, as well as adding or subtracting to the player's points.

#### MVPP Feature #6

Name: Background scrolling

Feature: The background automatically scrolls independent of the helicopter.

Constraints: The background doesn't scroll backwards. The helicopter cannot go off screen.

Scroll speed is capped so it moves slowly.

#### **MVPP Feature #8**

Name: Victim Generation

Feature: Victims spawn throughout the entire map for the player to rescue using the helicopter's movement feature illustrated in Feature #3

**Constraints:** The victims should spawn between a level of 5000 x 768 resolution, and should not clip through the floor or the ceiling of the level, but may clip from the front or to the back, ensuring that victims spawn within reach of the player, while still providing the rush of a challenge. The victims should scroll from right to left throughout the level so that the player-controlled helicopter may miss them.

#### **MVPP Feature #9**

Name: Environment Generation

Feature: Generate objects for the entire level, and make the objects scroll (1 pixel per frame).

 Listed all of our features (planned and/or implemented)

## About:

- Feature is described in simple, specific detail
- Constraints are listed to keep us in line with what to look for when testing.



# Design

Example Feature/Constraint(s):

## MVPP Feature #14

Name: Victim Distance Sensor

**Feature:** MiniAT has ports to return the X and Y distance from the helicopter to the nearest victim. The X distance is the distance between the front of the helicopter and the left side of the victim. The Y distance is the distance between the top of the helicopter and the helicopter and the bottom of the nearest victim.

**Constraints:** The victim which is found is the closest to the helicopter; the victim isn't behind the helicopter (already passed by);



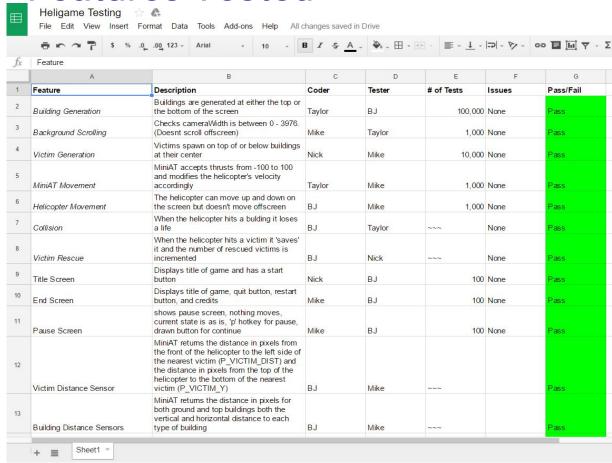


## **Features Tested**

	⊕ r ~ 7 s % .0	00 123 - Arial - 10 - I	B I 5 A	<b>→</b> - ⊞ - ∃	E - <u>1</u> -	P -	
f <sub>x</sub>							
	A	В	С	D	E	F	G
1	Feature	Description	Coder	Tester	# of Tests	Issues	Pass/Fail
2	Building Generation	Buildings are generated at either the top or the bottom of the screen	Taylor	BJ	100,000	None	Pass
3	Background Scrolling	Checks cameraWidth is between 0 - 3976. (Doesnt scroll offscreen)	Mike	Taylor	1,000	None	Pass
4	Victim Generation	Victims spawn on top of or below buildings at their center	Nick	Mike	10,000	None	Pass
5	MiniAT Movement	MiniAT accepts thrusts from -100 to 100 and modifies the helicopter's velocity accordingly	Taylor	Mike	1,000	None	Pass
5	Helicopter Movement	The helicopter can move up and down on the screen but doesn't move offscreen	BJ	Mike	1,000	None	Pass
7	Collision	When the helicopter hits a bulding it loses a life	BJ	Taylor			code reviewed
В	Victim Rescue	When the helicopter hits a victim it 'saves' it and the number of rescued victims is incremented	BJ	Nick			Untested
9	Title Screen	Displays title of game and has a start button	Nick	ВЈ			coded, untest
0	End Screen	Displays title of game, quit button, restart button, and credits	Mike	ВЈ			coded
1	Pause Screen	shows pause screen, nothing moves, current state is as is, 'p' hotkey for pause, drawn button for continue	Mike	BJ			coded
2	Victim Distance Sensor	MiniAT returns the distance in pixels from the front of the helicopter to the left side of the nearest victim (P_VICTIM_DIST) and the distance in pixels from the top of the helicopter to the bottom of the nearest victim (P_VICTIM_Y)	ВЈ	Mike			coded
3	Building Distance Sensors	MiniAT returns the distance in pixels for both ground and top buildings both the vertical and horizontal distance to each type of building	BJ	Mike			coded



## **Features Tested**



- List of features
- Descriptions
- ResponsibleCoder
- Tester
- Pass/Fail mark



# Methods of Testing

- Last cycle: more command line oriented
  - Features were more internal
- This cycle: graphical testing
  - Win/Lose/Start Screens
  - Music (Menu screen, in-game, end screen, and loss screen)
  - Victim Rescue



## Collision



# Rescuing



## MiniAT Sensors



	bj@arch
LOADed from P_GROUNDIS Address = 0x4020	Data = -1
LOADed from P_GROUNDIS Address = 0x4020	Data = -1
LOADed from P_GROUNDIS Address = 0x4020	Data = -1
LOADed from P_GROUNDIS Address = 0x4020	Data = -1
LOADed from P_GROUNDIS Address = 0x4020	Data = -1
LOADed from P_GROUNDIS Address = 0x4020	Data = -1
LOADed from P_GROUNDIS Address = 0x4020	Data = -1
LOADed from P_GROUNDIS Address = 0x4020	Data = -1

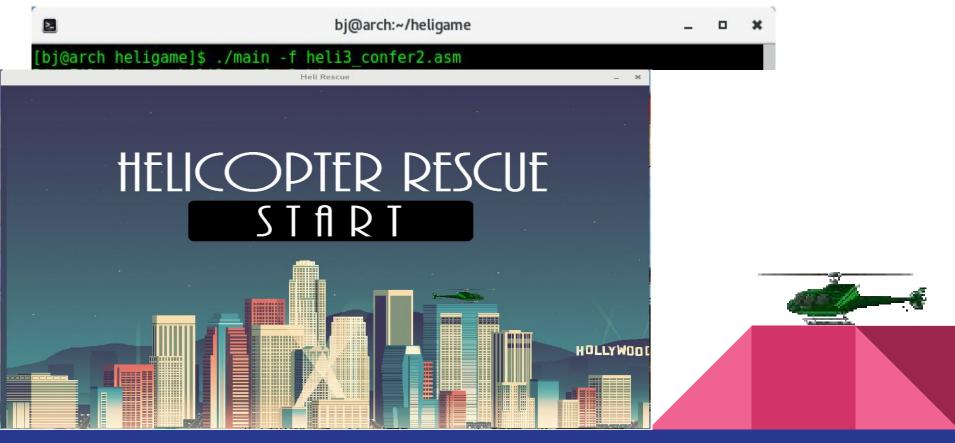
## MiniAT Sensors



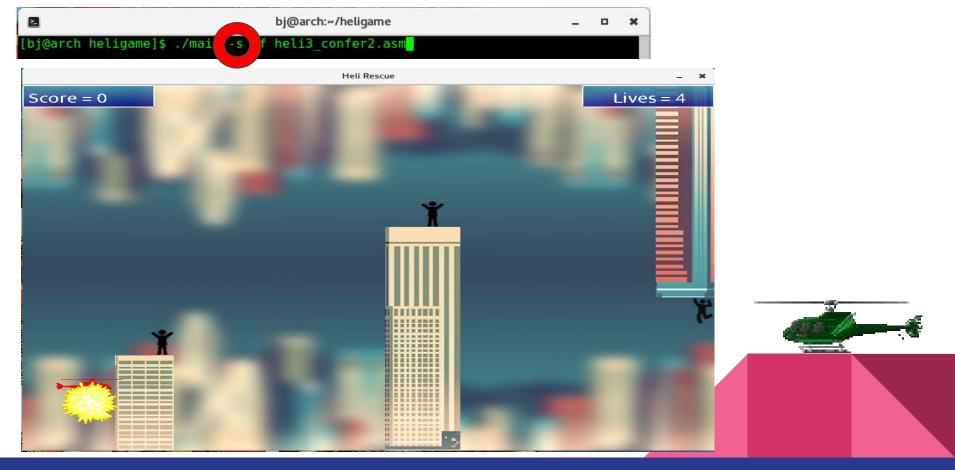
	bj@arch:~
LOADed from P_GROUNDIS Address = 0x4020	Data = 364
LOADed from P GROUNDIS Address = 0x4020	Data = 322
LOADed from P_GROUNDIS Address = 0x4020	Data = 280
LOADed from P_GROUNDIS Address = 0x4020	Data = 238
LOADed from P_GROUNDIS Address = 0x4020	Data = 196
LOADed from P_GROUNDIS Address = 0x4020	Data = 154
LOADed from P_GROUNDIS Address = 0x4020	Data = 112
LOADed from P_GROUNDIS Address = 0x4020	Data = 70

# **Testing Start Screen**

The most trivial of our screens.



# **Testing Start Screen**



# **Testing Life Counter**

```
Michael@DESKTOP-3V088VT MINGW64 ~/desktop/CS/C5370/testLives
$ grep "Hit! Lives = ^[0,1,4].*" testing.txt

Michael@DESKTOP-3V088VT MINGW64 ~/desktop/CS/C5370/testLives
$
Michael@DESKTOP-3V088VT MINGW64 ~/desktop/CS/C5370/testLives
$ grep "You Died!" testing.txt

Michael@DESKTOP-3V088VT MINGW64 ~/desktop/CS/C5370/testLives
```

 Lives is unsigned so it jumps to 4,294,967,295 instead of going negative.

```
livesBatch - Notepad
File Edit Format View Help
@echo off
echo Runs game 1,000 times
PAUSE
set loop=0
:loop
if "%loop%" == "1000" goto end
goto next
:next
START /WAIT PowerShell.exe ".\main.exe -fheli3.asm >>
testing.txt"
set /a loop=%loop%+1
goto loop
:end
echo STOP
```

## **Current Status...**

Our current game build:

v 1.1.0



## In Conclusion

-Were able to complete basic features for a relatively polished game, while meeting the standards set by our system and cycle intents.

-Missing many desired features Learned design/testing is more in depth than we realized

