

The background of the entire image is a scenic landscape. It features a calm lake in the foreground, reflecting the surrounding environment. In the middle ground, there are steep, rocky mountains with patches of green vegetation. The sky is a clear, pale blue with a few wispy clouds. The overall lighting suggests a bright, sunny day. A dark grey rectangular overlay is positioned in the center of the image, containing white text.

吉田学園情報ビジネス専門学校

ゲームスペシャリスト学科

Portfolio

東村 哲士

目次

ページ | オンライン | 全て表示 | 保留中 | ページに追加

ページー 6

自己紹介

1 ページ

マリオメーカー

1年作品:マリオメーカー

2 ページ~3 ページ

invaders のような game

2年作品:invaderつぼいgame

4 ページ~5 ページ

HAVOC
THE COUNTER ATTACK

2年作品:HAVOC(ステージエディタ)

6 ページ~7 ページ

その他学校でやった事

8 ページ~

自己紹介

東村 哲士#20001128

氏名

東村 哲士#20001128

ニックネーム

マサラ

好きなゲームジャンル

音楽ゲーム・シミュレーションゲーム

趣味

CoCTRPG・Vtuber

使用ソフト

Visual Studio 2015

PhotoShop2021

Maya

Spyder

Git

使用言語

C, C++, Python, VBA

マリオメーカー

マリオメーカー

ゲームで遊ぶ

ジャンル

2Dアクション

制作人数

1人

製作言語

C言語

製作期間

10~11月

ゲーム情報

メモ

マリオメーカーを再現！？

6種類のブロックと4種類の敵を使って
ステージを作ろう！

目次

自己紹介

マリオメーカー

マリオメーカー

invaders のような game

invaderっぽい
gameHAVOC
THE COUNTER ATTACKHAVOC
(ステージ
エディタ)その他学校
でやった事

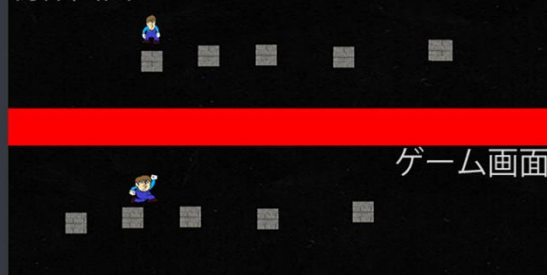
マリオメーカー

マリオメーカー

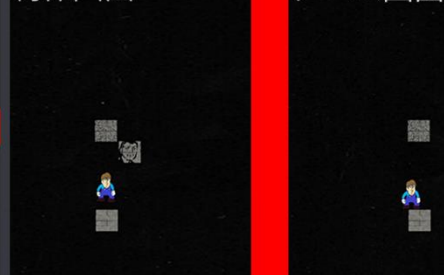
ここがマリオメーカーの説明の先頭です。

C言語でステージ制作ツールを
制作しました。

制作画面



制作画面



ゲーム画面

C言語でのツール作成で大変だったのは
「制作画面->ゲーム画面->制作画面」
と遷移した際にステージデータを保持
したままにする事です。

また、敵やブロックによって動きが
異なる物を作成しました！



蟹



魚

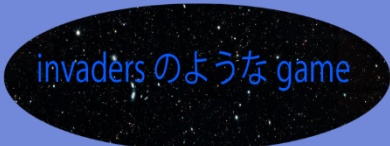


イカ



キモ猫

普通のブロック。
一番つまらない孔明ブロック。
孔明の罠「今です！」アイテムを出すブロック。
アイテムを出すと
普通に戻る悲しいやつちくわブロック。
親方！
上からちくわが！ヒビの入ったブロック。
下からの攻撃に
めっぽう弱いとげブロック(?)
触れるとチクッと
game over



invaders のような game

invaderのようなgame

ゲームで遊ぶ

ジャンル

2Dシューティング

制作人数

1人

製作言語

C++

製作期間

7~10月

ゲーム情報

メモ

敵・ボスを各4種類実装！

敵に応じて戦い方を変えよう！

また、ステージ作成ツールから制作！

簡単操作でステージを作成！

目次

自己紹介

マリオメーカー

マリオメーカー

invadersのようなgame

invaderっぽい
game

HAVOC
-THE COUNTER ATTACK-

HAVOC

(ステージ

エディタ)

その他学校
でやった事

invadersのようなgame

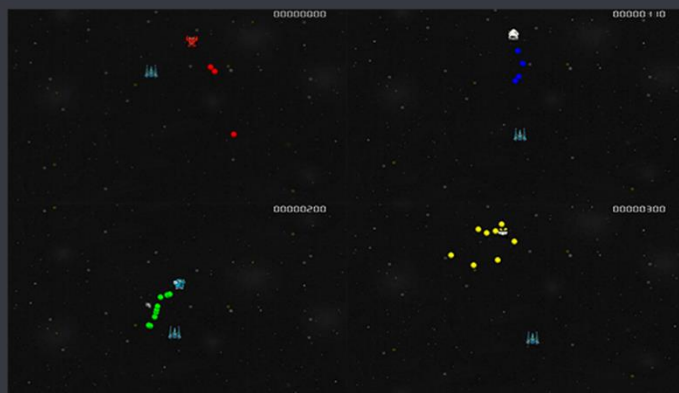
invaderのようなgame

ここがinvaderのようなgameの説明の先頭です。

C++で制作した初めてのゲームでした。

短い期間での制作でしたが、

敵・ボスの攻撃パターンをそれぞれ
制作しました！



また、ステージエディタから制作し、
ステージ情報を外部ファイルで管理
出来るようにしました。



```
460.000000
160.000000
0.000000
3
860.000000
160.000000
0.000000
3
660.000000
40.000000
0.000000
1
540.000000
280.000000
0.000000
1
780.000000
280.000000
0.000000
1
```



HAVOC -THE COUNTER ATTACK-

[ツールを見る](#)

ジャンル

3Dアクション

制作人数

6人

(ツール作成担当)

製作言語

C++

製作期間

10~11月

ゲーム情報

メモ

今回はチーム制作でツール担当でした。
チームメンバーと利便性を考えながら、
戻す・進む等のショートカットキーを実装
しました。また、モデルの情報を外部ファイル
で管理出来るようにしました。

目次

自己紹介

マリヲメーカー

マリヲメーカー

invadersのようなgame

invaderっぽい

game

HAVOC

HAVOC

(ステージ

エディタ)

その他学校

でやった事



HAVOC -THE COUNTER ATTACK-

ここがHAVOC -THE COUNTER ATTACK-の説明の先頭です。

今回のツールは得に力を入れて作成しました。

コピー・切り取り・ペースト・1つ戻る・1つ進める・

グリッド・X/Y軸固定等の29個のショートカットキーを

実装しました。

WASD : 移動
QE : 90度毎に向き変更
UJ : オブジェクトの高さ変更
QL : オブジェクトの座標を変更
1~0 : オブジェクト選択(1~10種類分)
テンキー1~0 : オブジェクト選択(11~20種類分)
Lcontrol + 1~0 : オブジェクト選択(21~30種類分)
Lcontrol + テンキー1~0 : オブジェクト選択(31~40種類分)
左クリック : オブジェクト消去
右クリック : オブジェクト設置
マウス : 視点操作
ホイール : 高さ変更
space : カメラの高さを上昇
Lshift : カメラの高さを下降
control + B : 変更内容をテキストファイルに書き出す
control + P : オブジェクトを全て消去
control + S : オブジェクト座標保存
control + C : オブジェクトの種類・向きを取得
control + V : 取得した情報で設置
control + X : 選択中オブジェクトを切り取り
control + Z : 操作を戻す
control + Y : 戻した操作を戻す
F1 : 選択中オブジェクトの座標を設置時の座標に戻す
F2 : 選択中オブジェクトの向きを設置時の方向に戻す
F3 : オブジェクトを設置するモードに変更
F4 : 敵を設置するモードに変更
X / Z : 軸固定
P : ポーズ切り替え

また、Xファイルやテクスチャの関連付けを外部ファイルで出来るようにしました。

```
##-----##
# 読み込むテクスチャ数                                # オブジェクト情報を一致させる
#-----#
NUM_TEXTURE = 8                                         #
#-----#
# 読み込むテクスチャファイル名                         OBJECTMATCH
#-----#
TEXTURE_FILENAME = NULL #0
TEXTURE_FILENAME = data/TEXTURE/building001.jpg #1
TEXTURE_FILENAME = data/TEXTURE/building101.jpg #2
TEXTURE_FILENAME = data/TEXTURE/building201.jpg #3
TEXTURE_FILENAME = data/TEXTURE/building301.jpg #4
TEXTURE_FILENAME = data/TEXTURE/building401.jpg #5
TEXTURE_FILENAME = data/TEXTURE/building501.jpg #6
TEXTURE_FILENAME = data/TEXTURE/ki.jpg #7
TEXTURE_FILENAME = data/TEXTURE/SolarPanel.jpg #8
#-----#
# 読み込むモデル数                                     OBJECTMATCH
#-----#
NUM_MODEL = 29                                          DRAW_NAME = ビル2
#-----#
# 読み込むモデルファイル名                             TEXTURE_FILENAME = 2
#-----#
MODEL_FILENAME = data/MODEL/building/building001.x #0  MODEL_FILENAME = 1
MODEL_FILENAME = data/MODEL/building/building101.x #1  TOUL_COLLISION_SIZE = 625 1650 625
MODEL_FILENAME = data/MODEL/building/building201.x #2  GAME_COLLISION_SIZE = 630 1650 630
MODEL_FILENAME = data/MODEL/building/building301.x #3  END_OBJECTMATCH
MODEL_FILENAME = data/MODEL/building/building401.x #4  OBJECTMATCH
MODEL_FILENAME = data/MODEL/object/Signal.x #6         DRAW_NAME = ビル3
MODEL_FILENAME = data/MODEL/object/Light.x #7         TEXTURE_FILENAME = 3
MODEL_FILENAME = data/MODEL/object/DustBox.x #8        MODEL_FILENAME = 2
MODEL_FILENAME = data/MODEL/object/Light2.x #9        TOUL_COLLISION_SIZE = 625 1650 625
MODEL_FILENAME = data/MODEL/object/wood1.x #10       GAME_COLLISION_SIZE = 630 1650 630
MODEL_FILENAME = data/MODEL/object/wood2.x #11       END_OBJECTMATCH
MODEL_FILENAME = data/MODEL/object/wood3.x #12       OBJECTMATCH
MODEL_FILENAME = data/MODEL/object/bench.x #13        DRAW_NAME = ビル4
MODEL_FILENAME = data/MODEL/object/tukue.x #14        TEXTURE_FILENAME = 4
MODEL_FILENAME = data/MODEL/object/guardrail.x #15    MODEL_FILENAME = 3
MODEL_FILENAME = data/MODEL/object/playground.x #16   TOUL_COLLISION_SIZE = 625 1650 625
MODEL_FILENAME = data/MODEL/object/signInter.x #17   GAME_COLLISION_SIZE = 630 1650 630
MODEL_FILENAME = data/MODEL/object/signNoPark.x #18  END_OBJECTMATCH
MODEL_FILENAME = data/MODEL/object/car.x #20
```

その他やったこと

- 0_Cascade.xml
- 1_Cascade.xml
- 2_Cascade.xml
- 3_Cascade.xml
- 4_Cascade.xml
- 5_Cascade.xml
- 6_Cascade.xml
- 7_Cascade.xml
- 8_Cascade.xml
- 9_Cascade.xml

ゲームで遊ぶ

ジャンル

AI

製作人数

1人

製作言語

Python

製作期間

約3ヵ月

ゲーム情報

メモ

授業外でPythonいじった！！！！

目次

自己紹介

マリヲメーカー

マリヲメーカー

invadersのような game

invaderっぽい
game

HAVOC
STEEL COUNTER ATTACK

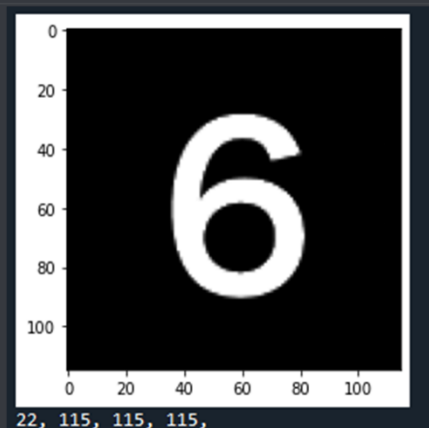
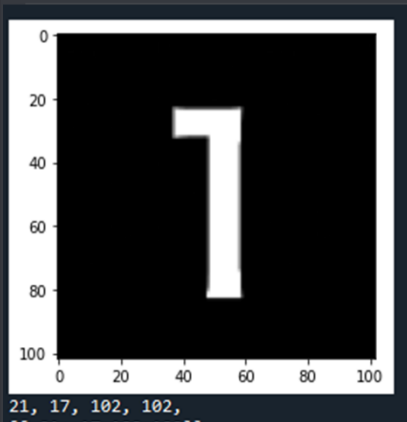
HAVOC
(ステージ
エディタ)

その他学校
でやった事



その他やったこと

ここがその他やったことの説明の先頭です。



自作のデータセットで、
カスケード分類器を
作成しました！

また、TensrowFlowでも分類器を作成
してみました！

```
0 Discrimination : 98.8821138211382 %
1 Discrimination : 99.29577464788733 %
2 Discrimination : 97.98270893371757 %
3 Discrimination : 98.42364532019704 %
4 Discrimination : 98.67482161060143 %
5 Discrimination : 98.31460674157303 %
6 Discrimination : 98.7460815047022 %
7 Discrimination : 98.73170731707317 %
8 Discrimination : 98.33852544132918 %
9 Discrimination : 98.31349206349206 %
```