

情報知能工学演習Ⅲ

個人課題レポート

提出日：平成27年5月28日

学籍番号：1405076T

氏名：平田 恭嘉

1. 追加機能概要

本課題では、サンプルとして与えられている鬼ごっこネットワークゲームに以下の機能を追加した。

- ワープオブジェクト(場所はランダム)
- 二人が合計 500 回動くと発生するサドンデスモード
- 逃げる側の人のみが使える、W、S、A、D キーを押すことによって上下左右に 2 マス進めるチート(回数制限あり。50 回)

これにより、ワープを用いて移動することで逃げ道が広がり楽しくなっている。ワープの位置は毎回ランダムなので毎回異なったステージが楽しめる。

またサドンデスモードによってステージが突然4分の1に縮小するのでいつまでも単調に鬼ごっこをすることなくハラハラ感が出てくる。

またチート機能によりなんとしてでも逃げ切ることができる。友達に内緒にして使うとおもしろい。ただ、無限に使えるとゲーム性が失われるので 50 回までという回数制限を設けた。これによりゲームとして成り立ちつつ笑える。

図 1 に、サドンデスモードになってステージが狭くなった様子を示す。



図 1 : クライアントにおける〇〇時のウインドウキャプチャ画面

2. 実装概要

2.1. ソースファイル群

開発したゲームのソースファイル群は以下のとおり。

1) ライブラリ

- i) `game_lib.h, game_lib.c` : サーバとクライアントの通信などを実現する関数群
- 2) サーバ
 - i) `server.c` : サーバ
- 3) クライアント
 - i) `client.c` : クライアント
- 4) その他
 - i) `Makefile` : ライブラリ, サーバ, クライアント実行ファイル生成

2.2. コンパイル方法

2.1 節で示すソースファイル群があるディレクトリにおいて, コマンドラインから以下を実行する.

```
% make
```

これにより, 同ディレクトリに, 実行ファイル `server`, `client` が生成される.

また, 以下の操作により, 実行ファイル, オブジェクトファイルを消去できる.

```
% make clean
```

3. 使用方法

3.1. 起動方法

実行方法は以下のとおりである. なお, `[]` はオプションとなる引数を表している.

- 1) サーバプログラムを実行する. 起動方法は以下のとおり.

```
% ./server [port]
```

ここで, `port` によりサーバが使用するポート番号を指定できる. なお, 省略するとデフォルトの 6000 番が利用されるものとする.

- 2) クライアントプログラムを実行する. 起動方法は以下のとおり.

```
% ./client server_name | server_IP [port]
```

ここで, “|” は, “または” を表す.

3.2. 操作方法

3.2.1. サーバ

主に矢印キーを使って操作する.

左矢印は左に 1 マス移動

右矢印は右に 1 マス移動

上矢印は上に 1 マス移動

下矢印は下に 1 マス移動

A は左に 2 マス移動

D は右に 2 マス移動

W は上に 2 マス移動

S は下に 2 マス移動

Q を押すとゲーム終了

3.2.2. クライアント

主に矢印キーを使って操作する.

左矢印は左に 1 マス移動

右矢印は右に 1 マス移動

上矢印は上に 1 マス移動

下矢印は下に 1 マス移動

Q を押すとゲーム終了

4. 動作検証

今回、開発したプログラムは、サドンデスモード時に若干心配がある。2 人の合計移動数が 500 に達した時にサドンデスが始まり、500 歩目に動いたほうが先にサドンデスエリア内に引き込まれるのだが、その時に相手側が 1 歩も動かなければ相手はそのエリア内に引き込まれることはない。つまりサドンデスエリアに入らない方法があるということである。そうなるとサーバ側がサドンデスエリアに立てこもることでゲームが終わらないという恐れがある。ただし、相手が 1 歩でも動けば相手もエリア内に引き込まれるので大概の場合ゲームが成立し、また、両者動き回することは必至で、しかも突然サドンデスが始まるのでサーバ側がサドンデスエリア外に立てこもることはほぼないと思われる。

5. 考察

サンプルプログラムを解読してオブジェクトの設置方法や動かし方、壁

の処理、サーバとクライアントの通信処理のやり方を理解できた。ワープ機能を追加しているうちにオブジェクトの置き方や文章表示などに対する理解が深まった。またそれによって、壁の作り方がわかり、サドンデス機能へとつながった。サーバとクライアントの移動回数を共有するために通信部分を用いたので、そのおかげで通信処理のやり方についての理解が深まった。結果サドンデス機能が完成した。操作の面でも何らかの改良を施したかったためチート機能を考えた。

ワープ機能の実現にあたっては、当初、ワープマスが消える不具合があったのだが、再表示することで解決できた。

サドンデス機能の 2 人の移動回数を計測してその合計を条件にするところで大変苦労したのだが、自分の移動回数を n 、相手の移動回数を m として、 n の値を相手の m に代入し、 $m+n$ の値を評価することで解決した。 n の値を相手に送信するために通信部分のプログラムを書くのに手こずったが、何とか出来た。

6. 自己評価

今回の課題にあたっては、演習時間以外に、およそ 6 時間取り組んだ。その内訳は 1 時間ワープについて、1 時間壁の作り方について、4 時間通信部分についてである。

また、よくわからない関数があったり .h ファイルが何なのかわからなかったので [1] を参考にした。getch とはなんだろうと思っていたが [1] を読むことで入力したキーを返す関数だということがわかった。 .h ファイルに関数が定義されていることもよくわかった。

もっと面白いゲームをつくりたかったが、なかなかシンプルなゲームとなってしまったのが悔やまれる。

謝辞

本課題のでの時間設定については、能勢氏（学籍番号 XXXXXXXXX）にアドバイスをもらった。通信部分については飯塚氏にアドバイスをもらった。

ここに記して感謝の意を表す。

参考文献

[1] 柴田望洋 著、“明解 C 言語入門編” ソフトバンククリエイティブ 2004 年