

# プログラミング言語実験・C言語 第4回課題レポート

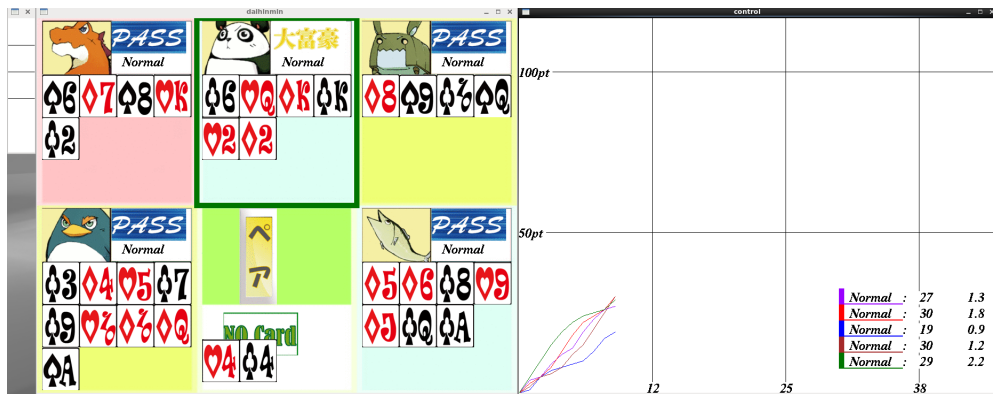
1510151 柳 裕太

2017 年 5 月 8 日

## 1 課題 7

### 1.1 スクリーンショット

図1 実際にペアが出された時のスクリーンショット



### 1.2 配列に対する操作

#### 1.2.1 make\_info\_table()

第一引数 info\_cards 配列を初期化したのち、各数字におけるカードの枚数 (第二引数 my\_cards 配列 1-3 行目各列の合計) を、info\_table 配列 4 行目に記録する。

#### 1.2.2 search\_low\_pair()

まず提出するカードを示す第一引数 dst\_cards 配列を初期化する。

その後、第二引数 info\_table 配列 4 行目を for 文で要素が 2 以上 (=ペアが存在する) の部分が存在するまでループを回し、もしペアが存在するならば、該当する列の第三引数 my\_cards 配列の各行要素を dst\_cards 配列に代入し、1 を返すことで処理を終了する。

もしペアが存在しないならば、0 を返すことで処理を終了する。

#### 1.2.3 select\_cards\_free()

まず、info\_table 配列を定義した後、make\_info\_table 関数を呼び出し、info\_table 配列と第二引数 my\_cards 配列を渡すことで、ペアの情報を取得する。

その後、count\_cards() 関数を呼び出し、第一引数 select\_cards 配列を渡すことで、出すカードが決定されている状態であるかを調べる。もし出すカードが決まっていなければ、search\_low\_pair() 関数を呼び出し、出せられるペアの中で最弱のものを提出カードに指定する。ここで search\_low\_pair() 関数にて出せられるペアが存在しないのならば、search\_low\_cards() 関数を呼び出し単騎カードを提出カードに指定する。

## 1.3 機能実装の決め手

### 1.3.1 make\_info\_table()

for 文により、my\_cards 配列 0 3 行目の任意の列の合計を info\_table 配列 4 行目の同じ列番号に代入することで、ペア情報の作成が可能となっている。

### 1.3.2 search\_low\_pair()

まず、for 文で info\_table 配列 4 行目にてペアを出せられる数字があるかを 1 行目から調べ、i 列に存在した時点で break している。これにより、その直後に if(ij=13) とすることで、ペアが存在しない = i が 14 (= ペアが存在すれば 14 に達する前に break されている) ケースは、ペアを指定するシーケンスをパスすることが可能となっている。

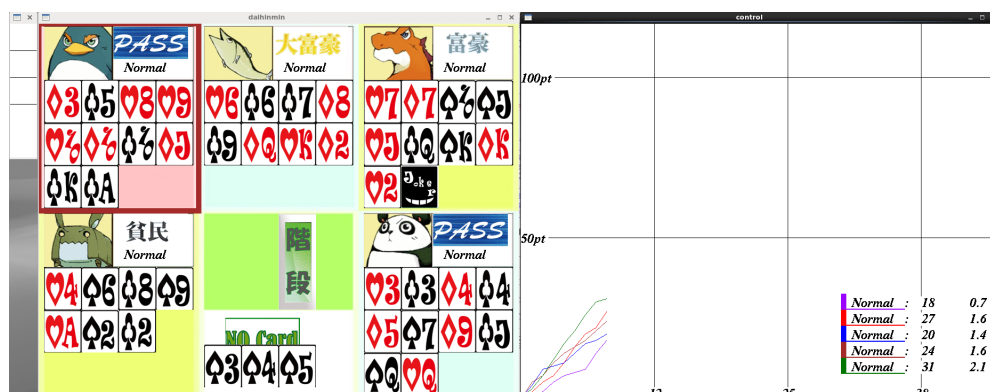
### 1.3.3 select\_cards\_free()

提出カードを指定するときに、search\_low\_pair() 関数或いは search\_low\_card() 関数を呼び出す前に、if(count\_cards(select\_cards)==0) を活用している。これにより、先に提出カードにペアが指定された後に、単騎提出カードに上書き指定されるのを防ぐことが可能となっている。

## 2 課題 8

### 2.1 スクリーンショット

図 2 実際に階段が出された時のスクリーンショット



### 2.2 配列に対する操作

### 2.3 機能実装の決め手