Objects

Card heap

Data members

```
typedef enum {heart, club, diamond, spade} color;
list<color> Cards;
```

Methods

```
bool ifempty(color * Cards);
void givecard(player p);
void init();//initilize the card heap, and shuffle
```

Player

Data members

```
const int capacity = 5;
unsigned int identifier;//the number of the player
vector<color> cards(capacity);
unsigned int score;
```

Methods

```
int* judge(color c);//check the usable card
color deal(int i);// deal the card
void update(unsigned score);//update the score
```

Table

Data members

```
int color_now;
vector<color> cards_now(2);
unsigned int offensive;//person number that deals
```

Methods

```
unsigned int cmp(vector<color>& cards_now);//compare the cards and give the result
unsigned int change_offensive();//change the offensive
void update_cards_now(color c);
```

The process

```
vector<player> p;
table t;
unsigned int p_now;
cardheap H;
H.init;
while(!H.ifempty()){
  p_now = H.change_offensive();
  update_cards_now(p[P_now].deal());

}
```