```
#ifndef _RATATAT_H_
#define _RATATAT_H_

#include <string>
#include <time.h>
#include <iostream>
using namespace std;

class Player;
class Pile;
class Card;

class RatATat
{
protected:
    unsigned int m_number_of_players; // number of players (also with the dealer include!)
    Player** m_players_arr;            //need to be array of pointers to player - so every organ wont be deleted!!! and with that i am solving the problem that player dont have default ctor - and i cant build array of players without giving the name to each player
    Pile* m_unused_cards_pile;         //because i'm  creating this pile in ctor and i dont want it will be deleted after ctor . i must do it with "new" , and "new" return pointer so: it must be "Pile* m_unsued_cards_pile;" , and not "Pile m_unsued_cards_pile;"
    Pile* m_thrown_cards_pile;

    //Extras:
    const static unsigned int m_min_number_of_players = 2;
    const static unsigned int m_max_number_of_players = 6;
    //Card* m_tmp_card_in_the_air;
    int m_whos_turn_indx;
    bool m_someone_shout_RatATat;
    Player** m_Players_with_min_hand_sum;   //for calculate who won!

public:
    RatATat() ; //ctor - creating array of pointers to players with size of maximum (6).also creating 2 piles : 1 new pile and shuffle it , and 1 empty for thrown cards
    ~RatATat();//dtor -     //Delete all created dynamic allocations:
    bool add_player(const string& name);  //if m_there_is_a_place_for_more_player = true , so : if name="Dealer" - create new AI_player , else create new Human-player . if there is no place for more player so prints that and return false.
    void play();

    //Extras:
    void card_distribution(); //distributing 4 cards to each player!
    void show_each_player_his_cards();    //for first time .show each player his card . then wait for pressing any key and clear screen!
    void play_turn();                           //play routine for all player .
    void player_pick_from_unused_cards_pile(); //picking card from unused
```

```
      cards pile and then continue to player choices what to do (throw / use)
    void player_pick_from_thrown_cards_pile(); //picking card from unused cards pile and then continue to player to use it! (he must use it)
    Card* get_card_from_unused_pile();         //just getting a card* from unused cards pile
    void throw_card_to_discard_pile(Card* card); //throw card to head of discard pile
    void print_top_card_of_discard_pile();
    void print_whos_turn();
    void player_do_your_move();                //each player turn rotuine
    void asking_player_from_what_pile_to_pick();
    //int player_choice_from_what_pile_to_pick(); //return 1 if player want to pick from unused pile , and 2 if he want to pick from discarded pile
    void player_set_ratatat_flag();
    void calc_players_hands();
    void who_won();
    int find_min_players_hand_sum();
    int get_number_of_players();
    int get_whos_turn_indx();

};

#endif // !_RATATAT_H_

```