```h
#ifndef _CARD_H_
#define _CARD_H_

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

class Player;
class Pile;
class RatATat;

class Card
{
protected:
    int m_value;          // The numerical value of the card
    string m_text;        //Describes the card "Cat/Rat,m_value" or for special_cards "Peek/Draw 2/Swap"

    //Extras:
    static int m_number_of_cards_already_made ;         //the number of cards that made till now   //??need to initialize it in cpp??//
    const static int m_total_number_of_cards_in_1_ratatat_pile = 54;    // total number of cards in 1 ratatat pile

public:
    //who that go to manage what card to create is pile! (first need to check with yael because if yes there is no need to toss_val , and if no so we need toss_val and class card will manage what card value to create)
    Card(const string& card_type , const int card_value );       //ctor - get card_type: (cat/rat/peek/draw 2/swap) , and card_value: if cat 0-5 , if rat 6-9 and if one of the special cards -1 . then create a card.
    //using default dtor
    //using default copy ctor
    //static , friends and virtual members and methodes:
    friend ostream& operator<<(ostream& os, const Card& c);  //prints card m_text
    //pure virtual methods:
    virtual void use(Player** players, int curr_player, RatATat& rat) = 0;  //there are some implementations : in play_card , and *each* of special_card!.
    virtual void print_card_action_menu() const = 0;   //there are 2 implementations - in play_card , and in special_card
    virtual int get_card_action_menu_max_num() const = 0;  //there are 2 implementations - in play_card and in special_card - always return 2

    //Extras:
    static int get_number_of_cards_already_made();                //  returns number_of_cards_already_made
    static int get_total_number_of_cards_in_1_ratatat_pile();     //return total_number_of_cards_in_1_ratatat_pile
```

```
38      int get_card_value() const ;
39      bool is_special_card() const;  //return 1 if card is special card and
          0 if not
40
41      //static void swap_cards(Card*& card_a_pointer, Card*&
          card_b_pointer);
42  };
43
44  ////////////////////////////////////////////////////////////////////////
      ///
45  class Play_card : public Card
46  {
47  protected:
48
49      //Extras:
50      static int m_Play_cards_hist_total[10];        //the total number of
          Play_cards from each type that need to be in 1 Ratatat pile. //can
          be const also?
51      static int m_Play_cards_hist_made_till_now[10];  //the number of
          Play_cards from each type that made till now.
52
53  public:
54      Play_card(const string& card_type, const int card_value); //ctor with
          empty implementaion - will get from initialization list of his sons
          - card_type = "Rat"/"Cat" and card_value (from toss_val of each of
          them), this will sent with the initialization list of himself to his
           father the members :m_value ,m_text
55      //using default dtor
56      //the virtual function of class card
57      void use(Player** players, int curr_player, RatATat&
          rat);             //does what he named about
58      void print_card_action_menu()
          const;                                  //does what he named about
59      int get_card_action_menu_max_num()
          const;                               //return 2 always
60  };
61
62  ////////////////////////////////
63  class Rat_card : public Play_card
64  {
65  protected:
66      static unsigned int m_num_cards; //static member describes how much
          Rat_cards made till now (need to initialize it in cpp in first lines
           of Rat_card) //?where the initializations?// //?where to put this
          line in public or protected?//
67
68      //Extras:
69      const static int m_total_Rat_cards_in_1_pile = 21;      //for new pile
70      const static int m_Rat_card_min_value = 6;              //for toss_val
71      const static int m_number_of_Rat_card_types = 4;        //for toss_val
72
73  public:
74      Rat_card(unsigned int card_value); //ctor with empty implementation -
```

```
        will send with initialization list to his father (the class
        play_card) the members m_text="Rat,m_value",and m_value=card_value
75                                      ////to create a card i will use
                    method toss_vall in class Rat_card to be card value .
76      //using default dtor
77      //static methodes
78      static int get_total_cards();  //return the number of Rat_cards made
        till now (return m_num_cards).  //?ask yael if thats what she
        wants!?//
79      static int toss_val();         //Scores a numerical value to the card
        according to the rules of 4 cards per card type
80
81      //Extras:
82      static int get_total_Rat_cards_in_1_pile(); //return the number of
        Rat_cards need to be in 1 pile (return
        total_Rat_cards_need_in_1_pile)
83  };
84
85  ////////////////////////////////
86  class Cat_card : public Play_card
87  {
88  protected:
89      static unsigned int m_num_cards; //static member describes how much
        Cat_cards made till now (need to initialize it in cpp in first lines
         of Cat_card) //??//
90
91      //Extras:
92      const static int m_total_Cat_cards_in_1_pile = 24;      //for new pile
93      const static int m_Cat_card_min_value = 0;              //for toss_val
94      const static int m_number_of_Cat_card_types = 6;        //for toss_val
95
96  public:
97      Cat_card(unsigned int card_value);   //ctor with empty implementation
        - will send with initialization list to his father (the class
        play_card) the members m_text="Cat,m_value",and m_value=card_value.
98                                          //to create a card i will use
                    method toss_vall in class Cat_card to be card value .
99      //using default dtor
100     //static methodes:
101     static int get_total_cards();  //return the number of Cat_cards made
        till now (return m_num_cards).  //?ask yael if thats what she
        wants!?//
102     static int toss_val();         //Scores a numerical value to the card
        according to the rules of 4 cards per card type
103
104     //Extras:
105     static int get_total_Cat_cards_in_1_pile(); //return the number of
        Cat_cards need to be in 1 pile (return
        total_Cat_cards_need_in_1_pile)
106 };
107 ////////////////////////////////////////////////////////////////////////
      ////
108 class Special_card : public Card
```

```
109 {
110 protected:
111
112 public:
113     Special_card(const string& special_card_type , const int card_value =
        -1);//ctor with empty implementaion - will get from initialization
        list of his sons - special_card_type = "Draw 2"/"Swap"/"Peek", this
        will initialize his father the
        members :m_text=special_card_type ,m_value =-1 ,m_text
114     //using default dtor
115     //the virtual function of class card
116     void print_card_action_menu() const;  //1 to discard card . 2 for use
        it
117     int get_card_action_menu_max_num() const;  //return 2
118 };
119
120 ////////////////////////////////
121 class Draw2_card : public Special_card
122 {
123 protected:
124     static unsigned int m_num_cards; //static member describes how much
        Draw2_cards made till now (need to initialize it in cpp in first
        lines of Draw2_card) //?where the initializations?// //?where to put
         this line in public or protected?//
125
126     //Extras:
127     const static int m_total_Draw2_cards_in_1_pile = 3;
128
129 public:
130     Draw2_card();   //ctor with empty implementaion - will send to his
        father (the class special card) with initialization list "Draw 2"
131     //using default dtor
132     //the virtual function of class card
133     void use(Player** players, int curr_player, RatATat& rat);
134     //static methodes:
135     static int get_total_cards();  //return the number of Draw2_cards made
         till now (return m_num_cards).  //?ask yael if thats what she
        wants!?//
136     //the virtual function of class card
137
138     //Extras:
139     static int get_total_Draw2_cards_in_1_pile(); //return the number of
        Draw2_cards need to be in 1 pile (return
        total_Draw2_cards_need_in_1_pile)
140 };
141
142 ////////////////////////////////
143 class Swap_card : public Special_card
144 {
145 protected:
146     static unsigned int m_num_cards; //static member describes how much
        Swap_cards made till now (need to initialize it in cpp in first
        lines of Swap_card) //?where the initializations?// //?where to put
```

```
        this line in public or protected?//
147
148     //Extras:
149     const static int m_total_Swap_cards_in_1_pile = 3;
150     void swap_cards(Card*& card_a_pointer, Card*& card_b_pointer);
151
152 public:
153     Swap_card();   //ctor with empty implementaion - will send to his
          father (the class special card) with initialization list "Swap"
154     //using default dtor
155     //the virtual function of class card
156     void use(Player** players, int curr_player, RatATat& rat);
157     //static methodes:
158     static int get_total_cards();  //return the number of Swap_cards made
          till now (return m_num_cards).  //?ask yael if thats what she
          wants!?//
159     //Extras:
160     static int get_total_Swap_cards_in_1_pile(); //return the number of
          Swap_cards need to be in 1 pile (return
          total_Draw2_cards_need_in_1_pile)
161
162 };
163
164 ////////////////////////////////
165 class Peek_card : public Special_card
166 {
167 protected:
168     static unsigned int m_num_cards; //static member describes how much
          Peek_cards made till now (need to initialize it in cpp in first
          lines of Peek_card) //?where the initializations?// //?where to put
          this line in public or protected?//
169
170     //Extras:
171     const static int m_total_Peek_cards_in_1_pile = 3;
172
173 public:
174     Peek_card();   //ctor with empty implementaion - will send to his
          father (the class special card) with initialization list "Peek"
175     //using default dtor
176     //the virtual function of class card
177     void use(Player** players, int curr_player, RatATat& rat);
178     //static methodes:
179     static int get_total_cards();  //return the number of Peek_cards made
          till now (return m_num_cards).  //?ask yael if thats what she
          wants!?//
180     //Extras:
181     static int get_total_Peek_cards_in_1_pile(); //return the number of
          Peek_cards need to be in 1 pile (return
          total_Draw2_cards_need_in_1_pile)
182
183 };
184
185
```

```
186 #endif // !_CARD_H_
187
```