,PROJECT DESCRIPTION AND MEMBER’S CONTRIBUTUION

“CASINO CARDS”

GROUP MEMBER’S DETAILS :

1.	Prateek sharma	(ECE) - 1910110285
2.	Mayank sharma	(ECE) - 19101101231
3.	Adhyan suri	(EEE) - 1910110026
4.	Satyaki Roychowdhury	(EEE) - 1810110221
5.	Meetendra Singh	(ECE)- 1910110232

PROJECT DESCRIPTION :

The following project is a set of 5 different types of card game . Every member has contributed with a game mode made by his own knowledge of data structures .Data structures used in the project are *stacks, structures and lists* .The common aspect of our project was to use randomly generated cards ,which every member followed in their respective modes .The project code consists of 4 header files , .c file and a “readme” file.

Details of the .h files:

1. instructions.h :

This header file consists of the rules and instructions to play a particular game selected by the user.
It is mostly only text and occupies a lot of space when put in the main program.

game_inst() : Instructions for the first game
game_inst_2() : Instructions for the second game
game_inst_3() : Instructions for the third game
game_inst_4() : Instructions for the forth game

----

2. display.h :

It consists of all the major functions that assist in the display attributes of the whole program. It includes messages to the user when they win or lose ('you win','you lose' etc.). It also consists the codes which is used to display the card(s). It also consists the functions which presents the selection menu, and score table on being called.

displayCard() : Used to display a single card.
displayCard_2() : Used to display two cards in a single row.
displayCard_3() : Used to display three cards in a row.
Display_list(req) : Used to display all the cards present in the list currently, one after another.

printTable(req) : A certain type of score card/table.
printTable_2(req) : Another type of score card/table.
displayMenu() : Used to present the selection menu; saves space
You_win() : Used to display 'You Win' message
You_lost() : Used to display 'You lost' message.
CARDS_GAME() : Used to display 'Casino Cards' on screen during the start.
card_game() : Used to display 'Card Games' on screen in the beginning of each game.

----

3. gameplays.h :

As the name suggests, it consists of all the gameplay functions of the program (except the Admin Mode). These functions take up a lot of space when spatially when included in the main c file. Also, keeping all the gameplay functions together in separately in a header file makes the code quite tidy and neat, thereby also convenient to examine how the various games function.

play_prateek(req) : gameplay function of the first game.
gameplay_mayank(req) : gameplay function of the second game.
playgame_satyaki(req) : gameplay function of the third game.
best_card_adhyan(req) : gameplay function of the forth game.

----

4. cards.h

It consists of the functions that create cards and their respective stacks and lists. It randomly allocates the values to various attributes of a card. Other functions included in this header file help creating the stacks and of the cards created by the previously mentioned function. Other miscellaneous functions like pop, insert etc. are also included in this header files.

createCards() : Used to create cards.
createStack(req) : Used to create stack of a definite number of cards.
pop(req) : Deletes the card present at the top of the stack of cards.
deleteNode(req) : Deletes the specified node(aka card), from a list of cards.
insert_List(req) : Insert a node(ie card), in the beginning of the list.
del_beg(req) : Delete the starting card of the list of cards.
del_search(req) : Delete the searched card(using a search key) from the list.
power_show(req) : Used to display the power attribute of all the cards present currently in the list.

Details of the .c files:

It contains the rest of the function which doesn’t no appropriately fall on any header file category And the main function also.
It has function which are not part of any game mode but certainly are used such as mypause, delay etc
Also included in this .c file

Delay(req) :Used to insert time between occurrence .
myflush(req)
mypause(req) :Used to a complete pause until user presses ENTER button to continue.
clrscr(req) : Used to clear the screen in output.
main() : Runs the code in a menu driven process

MEMBER'S CONTRIBUTION :

Note:- Each member has written their own contribution bellow.

1. Prateek Sharma :-

I made the first mode of the project i.e "PLAY WITH RANDOM CARDS" . I used "linked list" data structure in my mode .It is a comparison based game in random generated cards. It simply compares the value of property chosen by user to play with . My game function is 'play_prateek()' ,it is used to play game with random cards. It asks user to input any 1 property of card and then it compares it with corresponding value of computer's card ,One with more value gains a point.I introduced my group members about the coloring methods in the output . Also did work on del_search_key() function for "ADMIN MODE".

Functions made by Prateek :

1. play_prateek(card*,card*) - Responsible for running of the mode .
2. game_inst() - This function displays the instructions of game mode-1.
3. insert_list(card **) - It creates a linked list of 5 cards.
4. You_win() - It prints "YOU WIN" on the screen.
   You_lost() It prints "YOU LOST" on the screen.
5. cards_game() - It prints "CARDS GAME" on the screen.
6. myflush() - These two functions pauses the program and
   mypause() continues when the user press [ENTER].
   [Reference - Google]
7. clrscr() - Clears the output screen.

---

2. Mayank Sharma :-

Description of "Equal Hands Sum"

I am the guy behind the game stated above.

So basically this game is pretty similar to basic trading card games that we've been playing since our childhood.

This game is purely luck based and is played against the "system"(defined by us).

There are total of seven rounds in one game. And each player is given an opportunity of first pick of attribute in alternative chances.

The player who is supposed to go first has the option to pick a random attribute out of the given four attributes. After that the current score appears which is basically the value of the chosen attribute added to both the players' current score. Then the other player is given option to pick a random attribute out of the remaining three attributes. Then the value of the chosen attribute is added to the current score.

At the end of one round the one with the higher value of currrent score is given one point.

Therefore one with higher points at the end of the game is declared the winner.

<u>Functions used to play this game:</u>

gameplay_mayank(req.) : It is the major function for the whole gameplay of "Equal Hands sum".
game_inst_2() : Used to display the instructions to play this game
createCards() : Used to create cards which are further used to play various games.
createStack(req.) : Used to create the stack (imagine it as a deck of a card) for the players to play the game with.
pop(req.) : It is used to discard the card at the top of the stack as it has already been used to play.
myflush(stdin) : Misc. funtion.
mypause() : Pauses the screen till you press 'Enter'
clrscr() : Clears the screen above the line where its called.
You_win() : Displays 'You Win' message.
You_lost() : Displays 'You Lost' message.

<u>Functions made by Mayank :</u>

1. gameplay_mayank(req.) : It is the major function for the whole gameplay of "Equal Hands sum".
2. createCards() : Used to create cards which are further used to play various games.

3. createStack(req.) : Used to create the stack (imagine it as a deck of a card) for the players to play the game with.

4. pop(req.) : It is used to discard the card at the top of the stack as it has already been used to play.

---

3. Satyaki Roychowdhury :-

I have made the game called "Brain Game with cards". So basically in this game the player gets to see 3 cards and thereby has to answer 5 questions related to the same 3 cards. Out of these 5 questions 3 are MCQ’s and 2 are to be self written answer type questions. The player can enter the number of rounds he/she wants to play, and each round is played with different set of cards, and at the end of each round the current round score of the player is displayed.

I have used Stack data structure, to build this game. Depending on the number of rounds the player wants to play, the 3 cards are generated and inserted in 3 different stacks named by card a, card b, card c. In every round the respective cards are displayed, and at the end the respective three cards are deleted from the stack, and then the similar process is repeated for the other rounds as well.

Functions made by Satyaki :

1. playgame_satyaki(card**,card**,card**) - Resposible for running the game.

2. PrintTable_2(int p) - prints.the score of each round .

3. Create_cards() - used to generate random cards.

---

4. Adhyan Suri :-

I have made the game function of "LUCK OF AVERAGE" wherein the user is provided with a card with certain attributes(power, energy, karma and heal). The function will compare this card with a computer generated card and will tell us which card is better overall. This function considers all the attributes and calculates an average for both the cards(user and computer generated).
I have made this game mode by using structures. There will be 5 rounds for this game mode and the function will calculate the winner for all these rounds and declare a final winner.

Functions made by Adhyan :

1. Best_card_adhyan(card**,card**) - responsible for running the game

---

5. Meetendra Singh :-

I made the "ADMIN MODE". This mode is not exactly a type of game it is more like a program which uses application of linked list . It is a menu driven code .In this mode with the help of function 'create_cards() , power_functions()' i have generated five cards with random property values and displayed them on screen with display function . There i implemented 'del_search_key()' function to delete a card chosen by the user and then again print the cards but this time the deleted card will be gone . This is the implementation of deletion of node in linked list data structure . Here i would like to give credit to PRATEEK SHARMA who helped me during my struggle of coding for deleting the card successfully .

Besides that I have done some other small bits of works which includes creating all types of display functions of cards and “casino” display , compilation of all 5 different games into one single code and creating the main function with contribution of other member's code . I did all the small correction in final code to make the display presentable and good which contains inserting delay function between occurrence, shifting printing statements , colouring lines for enhancement ,writings instruction manuals, adding and deleting needed printf statements ,adding functions such as you_won(),you_lost(),delay(), mypause() at places where i felt it was needed in each members code , there were some glitches in other members games which i covered while my work . Plus in the text editor to make code look presentable and easy to understand i improved the alignment and spacing in the code .

Functions made by Meetendra :

1. Display_list(card*head) - Displays all the randomly generated card
2. Power_show(card*P) - Prints power of all five randomly generated cards.
3. del_search_key(card **h,int key) - Deletes the card with chosen power value.
4. displayCard_2(card *a, card*b) - Displays 2 card in a row.
5. displayCard_3(card *Card, card*card_2,card*card_3) - Displays 3 card in a row.
6. CARDS_GAME() - Displays ‘card casino’ at the beginning.
7. game_inst_2() ,game_inst_3() ,game_inst_4() - - Displays game instructions of “EQUAL HANDS SUM”, ”BRAIN GAME WIHT CARDS” and ”LUCK OF AVERAGE” .

****