**2015-2016**

SRINJOY SANTRA

CLASS XII

2015-2016



Annual Computer Project:

Game of UNO

**Design:**

This is an attempt to emulate the classic card game of UNO.In order to implement the game through programming in bluej, the fundamental concepts of Java language have been used.

The program has been divided into three classes;

1) Playcards2: (base class) manages cards beginning from card allocation, cards matching, display of discarded card and cards held by each player, deciding nature of power cards, keeping score of winner to sort cards once one card is discarded.

2) Game\_playUlt2: (derived class) starts the game, considers number of players (human and computer), effecting power cards, informing the required player the action of power card, selecting the appropriate card, drawing a card, passing a turn, deciding action of computer players.

3) File\_Display: displays wallpaper, rules of Uno, feedback details.

\*\* An added function of cheat code has also been included.When options of seeing and exiting is screened just enter “XLR8”. Enter cards in Block letters. e.g.WILD\_DRAW\_FOUR ; REVERSE BLUE etc. (This was made to check the proper functioning of the program at the micro level).

**System requirements:**

**Operating System: Windows XP/2007/2010**

**Java :Bluej environment (1.2 or higher version) compatible JDK(1.3 or higher version)**

**Contents:**

|  |  |  |
| --- | --- | --- |
| **Topics:** | **Page no:** | |
| **Java source code** | | **3** |
| **Method Description Table** | | **35** |
| **Variable Description Table** | | **38** |
| **Sample Output** | | **48** |
| **Acknowledgement-Bibliography** | | **49** |
| **Certificate** | | **50** |

**Method description table:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| ***Identifier*** | ***Access Specifier*** | ***Return Type*** | ***Purpose*** | ***Class*** |
| main() | public | void | to ask the user his choice; his choice of game; number of human players and computer players | Game\_playUlt2() |
| players() | public | void | to create cards for players; asks their name; decides the beginning card |
| gamerep() | public | void | to reverse the execution of control in a an array from ascending to descending and vice versa; make the next expected player that course of game has reversed |
| shortcut() | public | boolean | to carry the turn of a human player; to accept, draw a card, pass the chance; to check for “uno” calling from user at the end; to congratulate the winner |
| ***Identifier*** | ***Access Specifier*** | ***Return Type*** | ***Purpose*** | ***Class*** |
| shortcut() | public | boolean | to carry the turn of a computer player; to accept, draw a card, pass the chance(all by making random decisions) ; to check for “uno” calling; to declare the winner | Game\_playUlt2() |
| cheatcode() | public | void | to enable programmer to check the efficiency of the code; to increase chances of win |
| DRAWTWO  () | default | void | to implement action of draw two card |
| DRAWFOUR() | default | void | to implement action of draw four card |
| WILD  (Game\_playult2 ob, int x) | default | void | to implement action of colour changing by computer player or by user player |
| cardselect() | public | String | to generate a random card | Playcards2 |
| random() | public | int | to return a random number within range |
| ***Identifier*** | ***Access***  ***Specifier*** | ***Return***  ***Type*** | ***Purpose*** | ***Class*** |
| checkcom() | public | boolean | to check for matching cards | Playcards2 |
| display(String g[][],int cp[],int i) | public | void | to show cards a player holds |
| show() | public | void | to show required data of card |
| spclcard() | public | int | to effect proper functioning of power cards |
| score() | public | void | to calculate the score of the winner |
| exSelSort() | public | void | to selection sort the remaining cards after one card is played |
| masterprint() | default | void | to show wallpaper, uno rules, feedback | File\_Display |

**Variable description table:**

|  |  |  |  |
| --- | --- | --- | --- |
| class Game\_playUlt2() | | | |
| Identifier | Datatype | Purpose | Scope |
| ch | int | to accept a choice from user among ( new game ,…, exit) | main() |
| opt | int | to accept the type of game  (Player vs. Player ,…., classic mode ) |
| hp | int | to accept the no. of human players |
| co | int | to accept the no. of computer players |
| i | int | to ensure further details of rules are displayed |
| hp | int | to accept the no. of human players | players()  players() |
| co | int | to accept the no. of computer players |
| playerdata | int | to hold total no. of players |
| gamers [][] | String | array to store the cards each player holds during game |
| cardpresent [] | int | array to store no. of cards each player holds |
| name[] | String | array to hold names of each player |
| i | int | for-loop variable |
| s | String | to affirm continuation |
| j | int | for-loop variable |
| discard | String | to hold the initiating card |
| cad | int | to extract each token of a card |
| cardfunc[] | int | to store the power cards that are to be effected |
| position | int | to indicate the position of next expected player | gamerep()  gamerep() |
| begin | int | to continue till a player wins |
| limit | int | to pass positions  from one end to other |
| rev | boolean | denotes ascending or descending traversals |
| temp | boolean | to store the return value when the player misses chance to play |
| i | int | to store the adjacent position |
| j | int | to store the position of the other end |
| gamers [][] | String | array to store the cards each player holds during game |
| cardpresent [] | int | array to store no. of cards each player holds |
| name[] | String | array to hold names of each player |
| hp | int | to accept the no. of human players |
| cardfunc[] | int | to store the power cards that are to be effected |
| rev | boolean | denotes ascending or descending traversals | shortcut()  shortcut() |
| gamers [][] | String | array to store the cards each player holds during game |
| cardpresent [] | int | array to store no. of cards each player holds |
| name[] | String | array to hold names of each player |
| hp | int | to accept the no. of human players |
| i | int | to store the adjacent position |
| cardfunc[] | int | to store the power cards that are to be effected |
| drawx | boolean | to count whether theplayer has already drawn one card |
| limit | int | to store length of array name |
| s | String | to affirm continuation |
| option | int | to select a card from players’ pile |
| playerpos | int | to store the position of next player |
| u | int | to store the power card for next player |
| c | String | to affirm player’s uno calling |
| t | int | for loop variable | autocomp()  autocomp()  autocomp() |
| rev | boolean | denotes ascending or descending traversals |
| gamers [][] | String | array to store the cards each player holds during game |
| cardpresent [] | int | array to store no. of cards each player holds |
| name[] | String | array to hold names of each player |
| hp | int | to accept the no. of human players |
| i | int | to store the adjacent position |
| cardfunc[] | int | to store the power cards that are to be  effected |
| drawx | boolean | to count whether theplayer has already drawn one card |
| combination[] | int | to select the valid card options |
| y | int | to store a cardone at a time |
| x | int | to count number of valid cxards |
| flag | int | to find whether a valid random card is created |
| n | int | to store a random position |
| o | int | to store the element at the random position |
| playerpos | int | to store the position of next player |
| u | int | to store the power card for next player |
| c | String | to randomly decide computer’s uno calling |
| g | int | for loop variable |
| g [][] | String | array to store the cards each player holds during game | cheatcode() |
| cp [] | int | array to store no. of cards each player holds |
| j | int | for loop variable |
| user | String | to store user given string |
| gamers [][] | String | array to store the cards each player holds during game | DRAWTWO() |
| cardpresent [] | int | array to store no. of cards each player holds |
| playerpos | int | to store the position of next player |
| count | int | to count and give the next player two cards |
| gamers [][] | String | array to store the cards each player holds during game | DRAWFOUR()  DRAWFOUR() |
| cardpresent [] | int | array to store no. of cards each player holds |
| playerpos | int | to store the position of next player |
| count | int | to count and give the next player two cards |
| n | int | to randomly generate a colour or accept a colour | WILD(Game\_playult2 ob,int x) |
| col | String | to store the colour |
| x | int | to make the method act differently for computer or human player call |
| class Playcards2 | | | |
| Identifier | Datatype | Purpose | Scope |
| card | String | to create a card | cardselect() |
| number | int | to pick a random  number |
| n | int | to accept the upper limit to create a random number | random() |
| g | String | to accept a card information | split()  split() |
| opt | int | to return as per requirement |
| num | String | to store initial part |
| col | String | to store colour |
| cardval | String | to copy g |
| count | int | to keep track of number or colour extraction |
| w | String | to temporarily hold colour or number |
| x | int | for-loop variable |
| g | String | to accept a card information | checkcom() |
| d | String | to accept discarded card details |
| g[][] | String | to accept all cards of the players | display(String g[][],int cp[],int i) |
| cp[] | int | to accept the number of cards held by players |
| i | int | the player number |
| j | int | while loop variable |
| q | int | to count number of cards showed in previous rows |
| pres | int | to hold cards present for current player |
| cj | int | to copy value of j |
| g | String | to accept a card information | show()  show() |
| p | int | to ensure return of correct data |
| num | String | to store initial part |
| col | String | to store colour |
| y | int | for-loop variable |
| num | String | to store initial part | display(String g) |
| col | String | to store colour |
| y | int | for-loop variable |
| scard | String | to accept the card information | spclcard() |
| p | int | to select effective part of power card |
| name | String | to hold the name of winner | score() |
| g[][] | String | to accept all cards of the players |
| cp[] | int | to accept the number of cards held by players |
| posi | int | to store position of winner in the array |
| a | int | for-loop variable |
| b | int | for-loop variable |
| total | int | to calculate total score |
| c | int | to find category of power card |
| flag | boolean | to indicate card match | exSelSort()  exSelSort() |
| i | int | for-loop variable |
| dis | String | to store initial part of card |
| gamers [][] | String | array to store the cards each player holds during game |
| cardpresent [] | int | array to store no. of cards each player holds |
| discard | String | to hold the initiating card |
| cardfunc[] | int | to store the power cards that are to be effected |
| playerpos | int | to store the position of next player |
| Class File\_Display | | | |
| p | int | to store user option | masterprint() |
| txxt | String | to store each line of file |
| ho | String | to store choice |

**Acknowledgement:**

I would like to pay my gratitude to my parents, Mr Kasinath Santra and Mrs Saswati Santra for their support.

I am indebted to my school computer Science teacher, Mr Abhishek Das for approving the topic and advising on improvement possibilities.

**Bibliography:**

Computer Science with Java, Class XII, Sumita Arora.

**Certificate:**

This project has been successfully completed within .

Signature of School Teacher:

Signature of Visiting Examiner: