Design Tool: Pesudocode

1. First We create Structure for the Game: Dog Gambling Game

struct DogGamblingGame{

1.1 To store Name Declare Variable

1.2 To store Win Percentage Variable

1.3 To store Winner Price

}

2. We will Make Functions to Handle the Process:

2.1 Function for Game Menu

char GameStart();

2.2 Function for Displaying List of Dogs

void DogList();

2.3 Function for Storing Information

void DogInfo();

2.4 Function for Randamizing the Winning Percentage

void Randomizer();

2.5 Function for Giving a Random Result of

int DogRace();

2.6 Function for Declaring Winner

void DogRaceInformation();

2.7 Function for Storing Data of the Races

void RaceData();

3. Function for Game Menu:

3.1 Display Options given in assingment

3.2 Declare a Variable to store User Choice

3.3 Take User Choice

3.4 Return Value of Choice

char Choice; // Taking User Choice

scanf(" %c",&Choice);

return Choice; // Returing the Choice

4. Function for Displaying List of Dogs:

4.1 Display S.no Dog Name Payout and Win Percentage

4.2 Declare A loop to print all the Value from the struct

for(int i = 0; i < 9; i++)

4.3 Calling Structure.variable Name

printf("\*\*\*\*\*\*\*\*\*\* %d. %7s %4d to 1 %7d%%\n", i+1, Dog[i].Name, Dog[i].Payouts, Dog[i].WinningOdds);

5. Function for Storing Information:

5.1 Make a Array for Name of Dogs, Payout, Win Percentage

char Names\_of\_Dogs[9][32] = {"FurryPals","Boxers",

"Rottweiler","Poodles","Bulldog", "Beagles","Bulldogs",

"Retrievers","Shepherds"};

int Payouts[9] = {2,5,10,15,50,20,10,5,3}; // Storing Payouts

WinningOdds[9] = {40,10,8,6,1,4,8,10,13}; // Storing Win

5.2 Populate the value of all the Arrays

5.3 Declare a Loop to insert data into the Array

for(int i = 0; i < 9; i++)

5.4 Calling Structure.variable Name

strcpy(Dog[i].Name, Names\_of\_Dogs[i]);

Dog[i].Payouts = Payouts[i];

Dog[i].WinningOdds = WinningOdds[i];

6. Function for Randamizing the Winning Percentage:

6.1 Declare a Variable to use in random Function

6.2 Declare a for Loop which access the data from win Percentage Array and store it in Random

for(int i = 0; i < 9; i++){

for(int j = 0; j < Dog[i].WinningOdds; j++){ // Accessing the Random Dog to Race with

RandomNum[Random++] = i;

}

7. Function for Race Stimulation:

7.1 Use srand with timing starting from zero

7.2 Return value of Random Number

srand(time(0)); // Timing starting with Zero

int Random = rand()%100; // Returing Random Number

return RandomNum[Random];

8. Function for Declaring Winner:

8.1 We need to declare some Variable for Wager, Dog Pick, Result

8.2 We will call the Dog List Function

8.3 Ask user to Give the Wager

8.4 Ask user to Pick Dog for the Race

8.5 Decrement value of Dog pick by 1 since Array start from 0 and our S.no start from 1

8.6 Call Race Function and store it into the result since it return result

printf("\n%s WINS THE RACE\nYOU WON %d\n", Dog[DogPick].Name, Wager\*(Dog[DogPick].Payouts));

sprintf(buffer, "\n%s WINS THE RACE\nYOU WON %d\n", Dog[DogPick].Name, Wager\*(Dog[DogPick].Payouts));

8.7 Compare value of Dog pick with Result if equal

8.8 Display the Winner and Store the Winner for further use

8.9 Else Display the Winner and store the Lose of the Race

printf("\n%s WINS THE RACE\nYOU WON %d\n", Dog[DogPick].Name, Wager\*(Dog[DogPick].Payouts));

sprintf(buffer, "\n%s WINS THE RACE\nYOU WON %d\n", Dog[DogPick].Name, Wager\*(Dog[DogPick].Payouts));

9. Function for Storing the Data of Race:

9.1 Now Insert the Data of the race into a Variable

9.2 And Increment the counter so that poiner goes to next address

Data = Data; // Storing the results inside Data

Counter = Counter + 1;

10. Main Function:

10.1 We need to declare some Variable for Options, RaceData, counter, RandomNumber

10.2 We need to import the struct as an object

10.3 We will call Function that Display Dog Information

10.4 We will then call Randomizer Function

char Option;

char RaceData[1000][100];

int counter = 0;

struct DogGamblingGame Dog[9]; // Importing Structure of DogGamblingGame as Dog object

int RandomNumbers[100];

DogInfo(Dog); // Calling DogInfo

Randomizer(Dog, RandomNumbers);

10.5 Inside Do while Loop

10.6 Compare the user Input

10.7 If user choose to Gamble

10.8 Call Dog Race Information Function and insert value inside it

DogRaceInformation(Dog, RandomNumbers, RaceData[counter])

10.9 If user choose to See Race Result

10.10 We will print All RaceData using Loop

printf("%s",RaceData[i]);

10.11 If user choose to Exit

10.12 We Greet user and Exit the Program