#include <stdio.h>

#include <stdlib.h>

#include <string.h>

#include <time.h>

// Franklyn Gonzalez, last edited 03/06/2019

#define pause system("pause")

typedef struct dog

{

char name[10];

} DOG; // struct used to name off all the dogs in the race

DOG dog[9] = { "Darty","Barts","Jack","Ma","Leroy","Manky","Sound","Coke","Bobby"} ; //names of all the dogs

void menu(); //menu print

void gamble(); //gambling option

int odds(); // creating chances for Greg

main() {

char toggle = '\0'; // toggle used for menu

int add = 0; // variable used to add bets

double bet[100] = { 0 }; // bets placed

char orderResults[100] = { '\0' };

int \*gregMightWin = 0; //

do

{

menu();

while ((getchar()) != '\n');

//requests char input

toggle = getchar();

switch (toggle)

{

case 'G':

toggle = '\0';

gamble(bet, orderResults);

break;

case 'R': // prints results and amount of bet money used

for (int i = 0; i < 25; i++)

{

add += bet[i];

}

printf("The total amount of money gambled is %.2lf \n", add);

for (int i = 0; i < 50; i++)

{

printf("%s \n", orderResults[i]);

if (orderResults[i] == '\0')

{

i = 50;

}

}

toggle = '\0';

break;

case 'L':

exit(-1);

break;

default:

break;

}

} while (toggle != 'L');

}

void menu() // prints menu

{

printf("[G]amble\n");

printf("[R]esults of All Races\n");

printf("[L]eave the Dog Track\n");

}

int odds(int gregMustWin) {

srand(time(0));

gregMustWin = rand() % 100;

if (gregMustWin <= 40) {

return 0;

}

else if (gregMustWin <= 10) {

gregMustWin = rand() % 2;

if (gregMustWin == 0 )

return 1;

else if (gregMustWin == 1)

{

return 7;

}

else

{

return 9;

}

}

else if (gregMustWin <= 8) {

return 2;

}

else if (gregMustWin <= 6) {

return 3;

}

else if (gregMustWin == 1) {

return 4;

}

else if (gregMustWin <= 4) {

return 5;

}

else if (gregMustWin < 8) {

return 6;

}

else if (gregMustWin < 13) {

return 8;

}

else

{

return 9;

}

}

void gamble(double bet[], char orderResults[]) {

int gregMustWin = 0;

char select = '\0';

int winner = 0; //identifies winner

int i = 0; // a placeholder variable used to get information from random numbers

int count = 0; // counter to reach 9 dogs for the position they run in

int payout = 0; // the payout odds for winning

int dogSelect = 0; // requested input from user for the selection of dog he chose to win

printf("Please select a dog by number.\n"); // a selection of the dogs in the race

printf("1. Darty\n");

printf("2. Barts\n");

printf("3. Jack\n");

printf("4. Ma\n");

printf("5. Leroy\n");

printf("6. Manky\n");

printf("7. Sound\n");

printf("8. Coke\n");

printf("9. Bobby\n");

scanf("%i", &dogSelect);

printf("How much would you like to bet?\n");

scanf("%lf", &bet);

printf("LET THE RACE BEGIN!\n ");

pause;

char orderRace[9] = { '\0' }; // creating the order of the dogs from first to last

// used to prevent sequence repetition

for (int j = 1; 0 < j; j--)

{

odds(gregMustWin);

if (gregMustWin == 9) //

{

i = rand() % 9;

}

count++;

if (count == 1 && (strcmp(dog[dogSelect - 1].name, dog[gregMustWin].name) == 0))

{

winner++; // comparing strings with the one Greg selected

}

if (count == 1)

{

strcpy(orderRace, dog[i].name); // orderRace copies the dog in first place

}

}

if (winner == 1) { // Greg won!

printf("And the winner is.....\n");

pause;

printf("%s \n", dog[dogSelect - 1].name);

printf("Congratulations! You won the pay out Greg! A prize will be waiting for you after this race...\n\n\a");

pause;

}

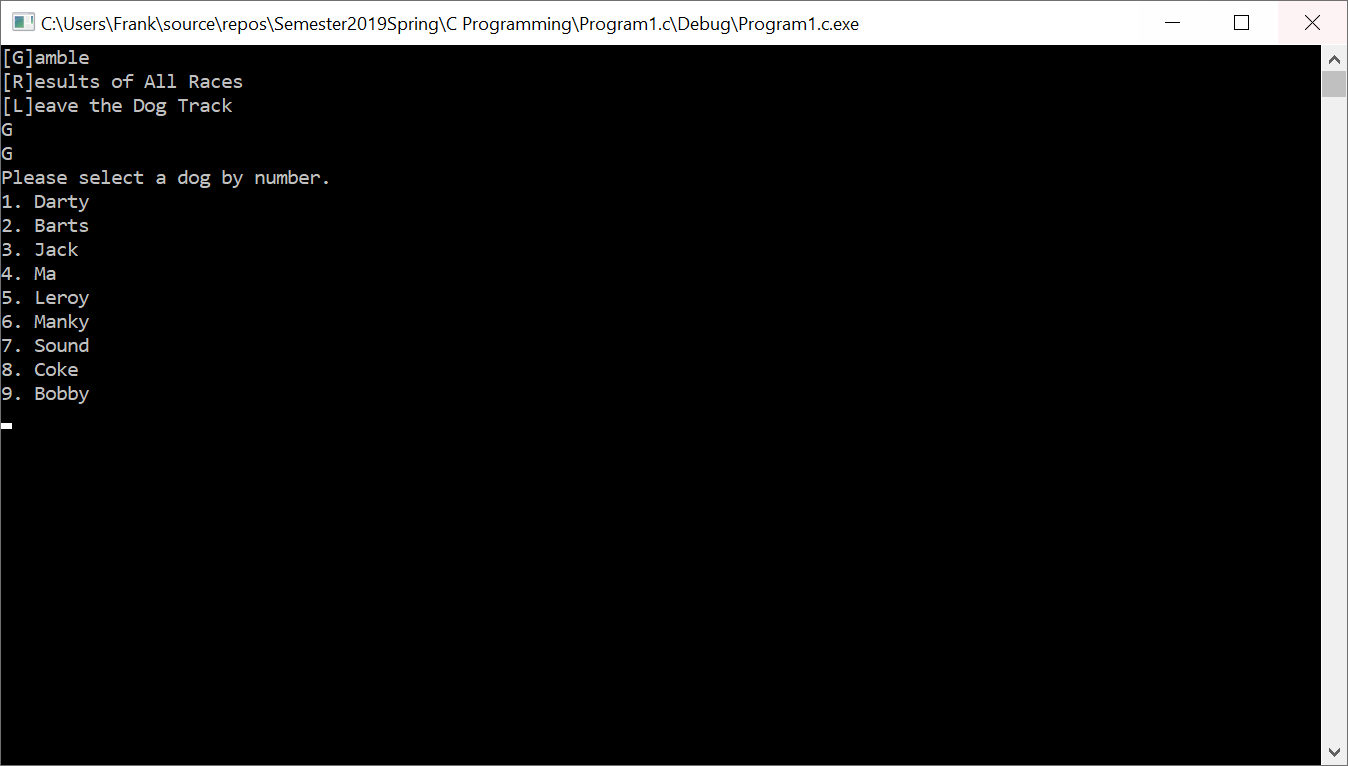
else // Greg didn't win

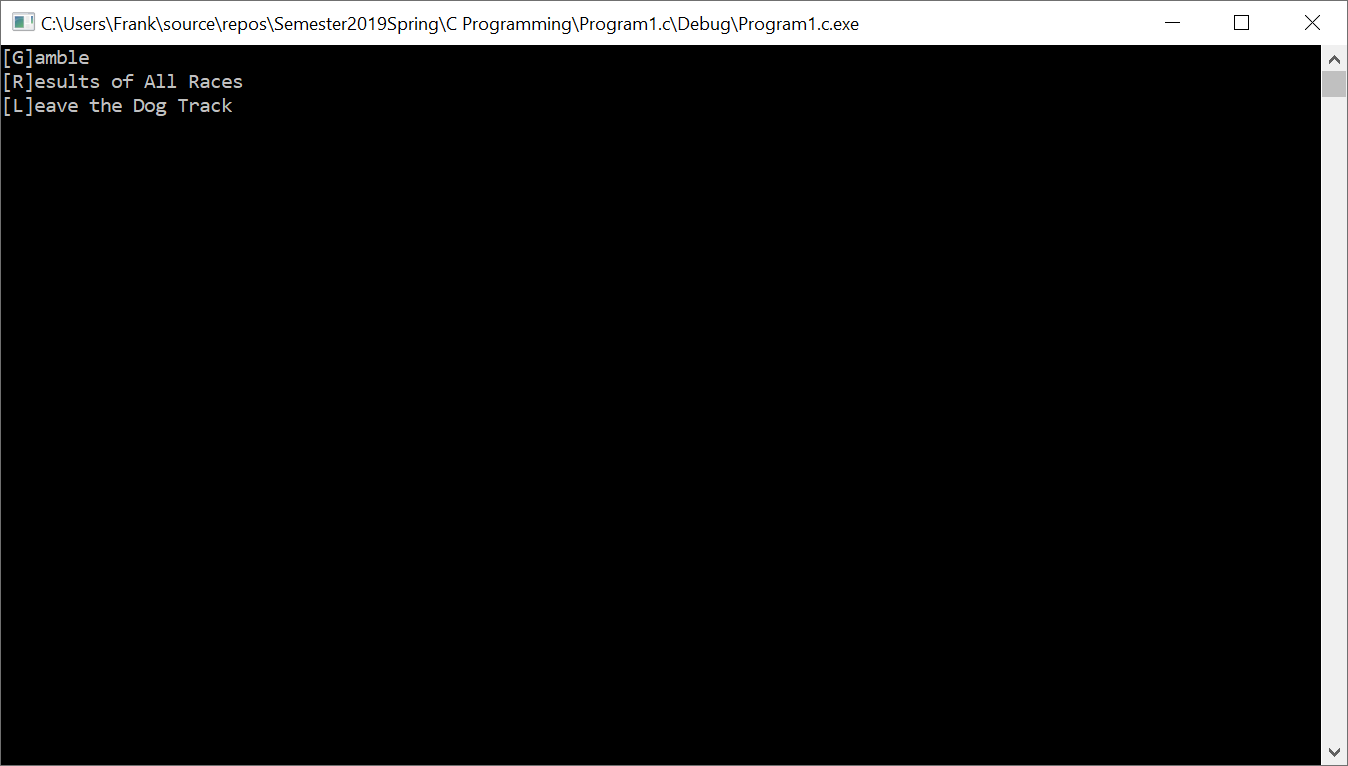
{

printf("Sorry.... but you didn't make the cut! Try again next time.\n");

}

}





|  |  |  |
| --- | --- | --- |
| **Input**  menu  odds  gamble  toggle  add  bet[100]  orderResults[100]  \*gregMightWin  gregMustWin  select  winner  count  payout  dogSelect | **Processing**  🡪 provides user selection🡪  🡪creates odds for Greg🡪  🡪makes a selection for Greg🡪  🡪takes user input🡪  🡪adds up Gregs’ bets🡪  🡪holds on to Greg’s current bet🡪  🡪saves the previous race winner🡪  🡪creates a gamble for Greg’s odds🡪  🡪sets Gregs gamble🡪  🡪takes the total pay that an employee has earned🡪  🡪finds a winner for Greg🡪  🡪counts the position each dog in is🡪  🡪creates a payout for Greg🡪  🡪allows Greg to select a dog 🡪 | **Output**  Displays menu  Finds the odds of winning out of 100 for Greg  Allows Greg to gamble on a dog  Places option  Displays total of Greg’s spendings  Saves greg’s bet  Displays the dog’s who won in the recent races  Returns the bet Greg made on a dog  Returns the bet Greg made on a dog  Takes input from Greg  Finds the dog Greg chose  Counts the position each dog is in  Allows Greg to make a payout  Takes Greg’s input |