void cleanBuffer() { - объявление функции для очистки буфера ввода

int a; - создаю переменную, чтобы сохранить в ней символы

while ((a = getchar()) != '\n' && a != EOF); - объявляю цикл,в котором getchar читает один символ и

присваивает его переменной а. дальше идет проверка,что этот символ не является символом новой строки или

концом файла

}

void user() { - объявляю функцию для режима игры где угадывает число сам человек

srand(time(NULL)); - активирую генератор случайных чисел текущим временем,чтобы каждый раз выводилось новое случайное число

int randomNum = rand() % 1000 + 1; - ввожу переменную рандом числа, беру его остаток от деления на 1000 и прибавляю 1,чтобы число было не меньше 1,но не больше 1000, так как это указано в условии игры

int znum, pop = 0; - ввожу переменные для хранения числа,которое вписывает человек,чтобы угадать загаданное и для счета попыток угадывания

printf("You guess the number\n I have thought of a number from 1 to 1000. Try to guess!\n"); - вывод начального текста,для объявления смысла игры

do {

pop++; - увеличиваем количество попыток на 1

printf("Enter your guess:\n"); - запрашиваем у пользователя ввести число,чтобы угадать загаданное

cleanBuffer(); - очищаем буфер ввода нашей функцией

scanf("%d", &znum); -чтение вводимого числа в переменную znum

while (znum < 1 || znum > 1000) { - объявляю цикл проверки вводимого числа

printf("Error. Enter a number from 1 to 1000\n");

cleanBuffer();

scanf("%d", &znum); - если число не подходит по условиям, то выводится текст о ошибке,после очищается буфер и снова вводится новое число пользователем.Цикл продолжается, пока игрок не введёт корректное число

}if (znum < randomNum) {

printf("The hidden number is higher\n");

}else if (znum > randomNum) {

printf("The hidden number is smaller\n");

}else {

printf("Congratulations! You guessed the number %d in %d attempts!\n\n", randomNum, pop); - в этих строчках кода мы сравниваем вводимое число с загаданным и пишем игроку его число больше или меньше загаданного. А если игрок угадал, то выводим поздравление, загаданное число и количество попыток

}

} while (znum != randomNum); - этот цикл повторяется пока игрок не угадает число

}

void computer() { - функция для режима игры где число угадывает компьютер

int left = 1, right = 1000; - вводим переменные границ поиска ( для бинарного поиска)

int znum, pop = 0; - переменные для догадки компьютера и подсчета попыток

int num = 0; - перемнная для загаданного числа человеком

char sym = 0; - символ для ответа игрока

printf("The computer guesses the number\n");

printf("Think of a number from 1 to 1000. I'll try to guess it!\n");

cleanBuffer();

scanf("%d", &num); - вывод информации правил игры,очистка буфера ввода и чтение загаданного числа

while (num < 1 || num > 1000) { - ввожу цикл для проверки числа как и в прошлой функции

printf("Error. Enter a number from 1 to 1000\n");

cleanBuffer();

scanf("%d", &num);

} -- если число не подходит по условиям, то выводится текст о ошибке,после очищается буфер и снова вводится новое число пользователем.Цикл продолжается, пока игрок не введёт корректное число

printf("Enter:\n");

printf("'>' - if your number is higher than mine\n");

printf("'<' - if your number is less than mine\n");

printf("'=' - If I guessed right\n");

выводим объяснения для игрока,что вводить в какой ситуации (инструкция)

do {

pop++; - увеличиваем количество попыток на 1

znum = left + (right - left) / 2; - методом бинарного поиска находим середину диапозона между границами

printf("My guess %d\n", znum);

cleanBuffer();

scanf("%c", &sym);

компьютер выводит свою догадку, очищается буфер ввода и читается ответ игрока ( загаданное число меньше,больше и равно предположению компьютера)

if (sym == '>') {

left = znum + 1;

}else if (sym == '<') {

right = znum - 1;

}else {

printf("Hooray! I guessed the number %d in %d attempts!\n\n", znum, pop);

} – если загаданное число больше, то сдвигаем левую границу. Если число меньше,то сдвигаем правую границу. Все по методу бинарного поиска. Если число равно загаданному, то поздравляем и выводим число и количество попыток

} while (sym != '='); - цикл продолжается пока компьютер не угадает число

}

int start() { - объявляем функцию для выбора режима игры,которая возвращает число

int game = 0;

printf("Enter the game mode (1 or 2)\n 1 - the program guesses a number\n 2 - the user guesses a number\n other sym - for exit\n");

scanf("%d", &game);

return game;

создаем переменную режима игры, выводим текст с инструкцией для выбора режима,считываем и возвращаем число,которое означает выбор режима

}

int main() { - объявляем основную функцию,где «собираем» по кусочкам из других функций игру

int mode = start(); - вводим переменную,которая имеет вид числа возвращаемого из функции про выбор режима игры

if (mode != 1 && mode != 2) {

return 0;

}

Проверяем корректность числа для выбора режима,если это не 1 и не 2,то завершаем работу программы

while (mode == 1 || mode == 2) { - объявляю цикл, где в зависимости от числа возвращаемого функцией start запускается соотвествующий режим игры

if (mode == 1) {

user();

}else if (mode == 2) {

computer();

}

cleanBuffer(); - очистка буфера и после запрос нового режима игры

mode = start();

}

return 0; - завершение программы

}