Comecocos:

Realiza la función de titulo() con el archivo fant.txt. Empieza la primera ronda, el codigo continua en la función inicializar() donde copia el tablero t[][], de un archivo llamado tablero.txt e inicializa todos las variables. Comienza la partida se sitúa el Pacman y los 4 fantasmas en sus sitios correspondientes, y si el usuario pulsa una tecla se moverá el Pacman sino, continua el juego y si se mueven los fantasmas. Si pulsas una tecla te lleva a la función mov() que dependiendo que pulses mueve el fantasma hacia un lado o te lleva a menu(). El Pacman se puede mover por el tablero pero no te deja moverte hacia donde haya paredes. El funcionamiento de los fantasmas: cada fantasma tiene cuatro movimientos, un vector con cuatro valores representa cada movimiento si un lado tiene pared cambia el valor del vector correspondiente a 0. Después con un rand() se coge un valor aleatorio del vector y si es cero vuelve a coger otro aleatorio porque el fantasma no puede ir en esa dirección y si no lo es mueve al fantasma hacia ese lado. También es importante notar colafant[] lo que hace es guardar el carácter que hay en el tablero antes de que un fantasma se lo coma para volver a ponerlo una vez pase el fantasma y no se borre. Al principio hay dos fantasmas metidos en el rectángulo de en medio que no pueden salir hasta que pasen 15 segundos, cuando se abrirán las puertas. Si te comes un ‘+’ que es una fruta el símbolo de los fantasmas cambia de 178 a 177 y durante 10 segundos te los puedes comer y aumentar la puntuación. La ronda continuará hasta que te comas un fantasma cuando no debes (tus coordenadas están guardadas en i, j entonces si t[i][j]==178 pierdes), te comas todos los ‘.’ del tablero o en el menú de pausa le des a terminar. Cada ronda va más rápida que la anterior. Al final compara tu puntuación con la que el highscore guardado en puntos.txt y si es mayor te lo dice y guarda la nueva puntuación como highscore.

Tetris:

Comienza realizando las funciones titulo() leyendo tetris.txt e inicializar. Empieza el juego, esta todo metido en un while(b==0). b cambia al valor 1 si las figuras llegan hasta arriba o si en el menú de pausa eliges dejar el juego. Aleatoriamente escoge una figura y un lugar de salida usando dos rand. Después si pulsas una tecla va a mov() para leer el movimiento y después va a la función figuras() para validar el movimiento según la figura. Observando si la figura dependiendo de su forma choca con otra. Si no choca continua bajando y, si choca, deja la figura donde está y empieza el proceso con otra. Cuando una línea entera vale 219 (símbolo de los pixeles de la figura), se borra, todas las líneas por encima de esta copian a la de encima y te suma un punto. Al finalizar el juego, si tu puntuación es mayor que la que está guardada como highscore en highscore.txt, guarda la nueva, sino, se queda con la antigua.

Blackjack:

Imprime el titulo leyéndolo del archivo imágenes.txt en la función titulo(). El juego comienza preguntando cuantos jugadores van a participar, después pregunta si quieren jugar una partida anterior (guardada en el archivo saved.txt) o empezar una nueva y te pide el dinero que tiene cada uno. Después empiezan las rondas, cada ronda te pregunta cuanto quieres apostar y reparte dos cartas a cada jugador, el número de las cartas se guarda en card[][] y el palo en palo[][]. Para imprimirte las cartas las imprime por líneas para que se vean todas seguidas y no una debajo de la otra. Te pregunta si quieres otra hasta un máximo de ocho cartas y compara valor[j] que guarda cuanto suman tus cartas con el valor[numero] de la casa. Dependiendo de los dos valores y según las normas del juego te dirá si ganas o pierdas y te duplicara o te restara tu dinero. Después de cada ronda te pregunta si quieres jugar otra. Si la respuesta es no no, guarda la puntuación de cada jugador en el archivo saved.txt para continuar en otro momento.

Buscaminas:

Imprime el titulo leyéndolo del archivo texto.txt desde la función titulo(). Continua inicializando todas las variables en la función inicializar(). Comienza la partida, el código pregunta al usuario cuantas bombas quiere poner en el tablero de 10x10. Después, aleatoriamente, pone ese número de bombas por el tablero. Mira cuantas bombas rodea a cada casilla y guarda cada valor en su hueco correspondiente en la matriz bombas[][]. Imprime el tablero usando la matriz vacio[][], en la que todos los huecos valen ‘-‘ y por la que el jugador se puede mover hasta pulsar intro o tab. Una vez pulsa intro se iguala la matiz vacio con la matriz bombas en el hueco elegido por el usuario. Si pulsa tab pondrá el carácter ‘¶’ simulando una bandera en el lugar elegido. Este proceso continuará hasta que el jugador pulse intro en una bomba, en el menú de pausa elija terminar el juego o ponga banderas en todas las bombas.

Snake:

El programa empieza imprimiendo el título leyéndolo del archivo snake1.txt desde la función tirulo(). Continua inicializando todas las variables en la función inicializar(). Comienza la partida, y aparecen la serpiente y la manzana aleatoriamente por el tablero. Te puedes desplazar utilizando las flechas y cada vez que las coordenadas de la cabeza sean las mismas que la de la manzana crecerá el tamaño de la serpiente usando un realloc. Si las coordenadas de la cabeza de la serpiente son iguales a las de cualquier punto de la cola, pierdes una vida. El juego termina cuando pierdes las tres vidas que tienes o en el menú de pausa elijes no continuar el juego. Finalmente te dirá si has obtenido un nuevo highscore comparándolo con el guardado en el archivo highscore.txt y , en caso de superar el récord, guardará el anterior en el archivo backup.txt