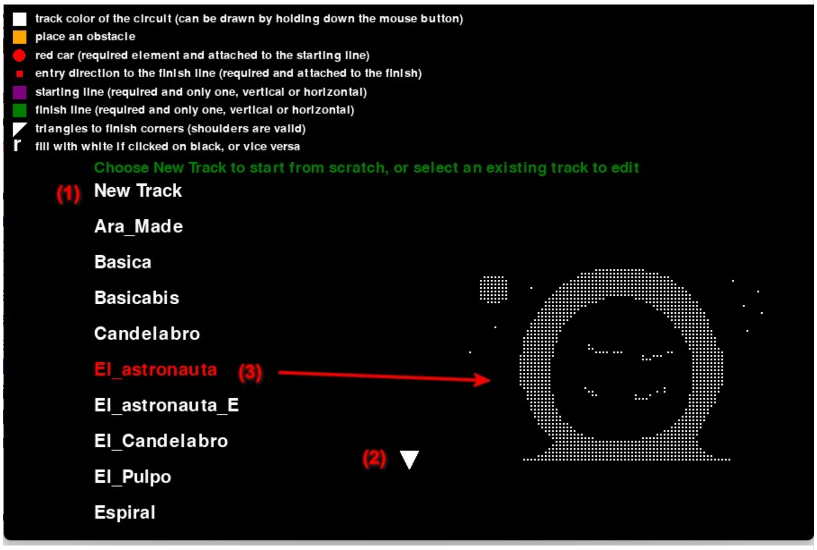
Track Editor (the inside of F1-RaceTrack)

The Track Editor is a utility that allows us to create a new race track for the game or modify an existing one. The tracks we can design are only limited by our imagination but can basically be divided into two categories: those where the start and finish are at different points defining an open circuit - you start from the start, reach the finish and the race ends when you cross it. There are also closed circuits where the start and finish are connected, and therefore races can be several laps.

The initial screen presents a reminder of the use of the tools and allows us to choose whether we want to create a new one by choosing the first option: **(1)** New Track or edit one of the existing ones. If the already created tracks don't fit on the screen, arrows **(2)** will indicate this. We can choose the track with the keyboard arrows and **ENTER** or with the mouse (the mouse wheel or the keyboard arrows allow us to view the tracks below or above the arrows). There is also a thumbnail of the selected track **(3)**.

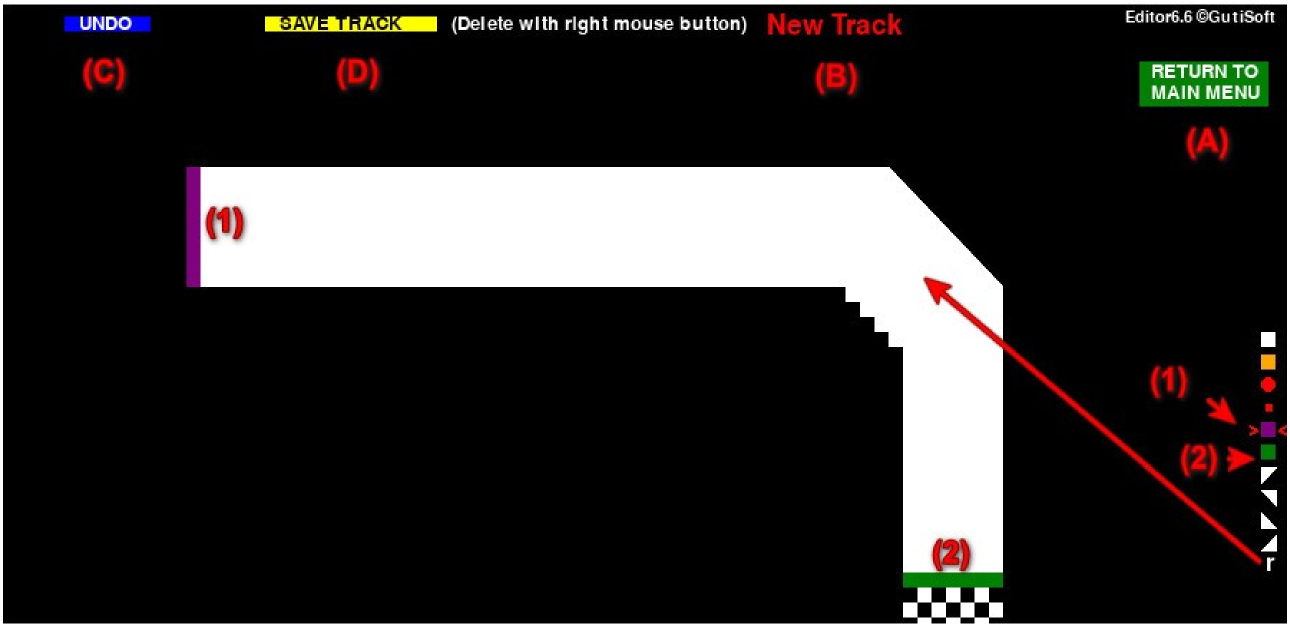


□ track color of the circuit (can be drawn by holding down the mouse button)
🟧 place an obstacle
● red car (required element and attached to the starting line)
▪ entry direction to the finish line (required and attached to the finish)
🟪 starting line (required and only one, vertical or horizontal)
🟩 finish line (required and only one, vertical or horizontal)
◸ triangles to finish corners (shoulders are valid)
r fill with white if clicked on black, or vice versa

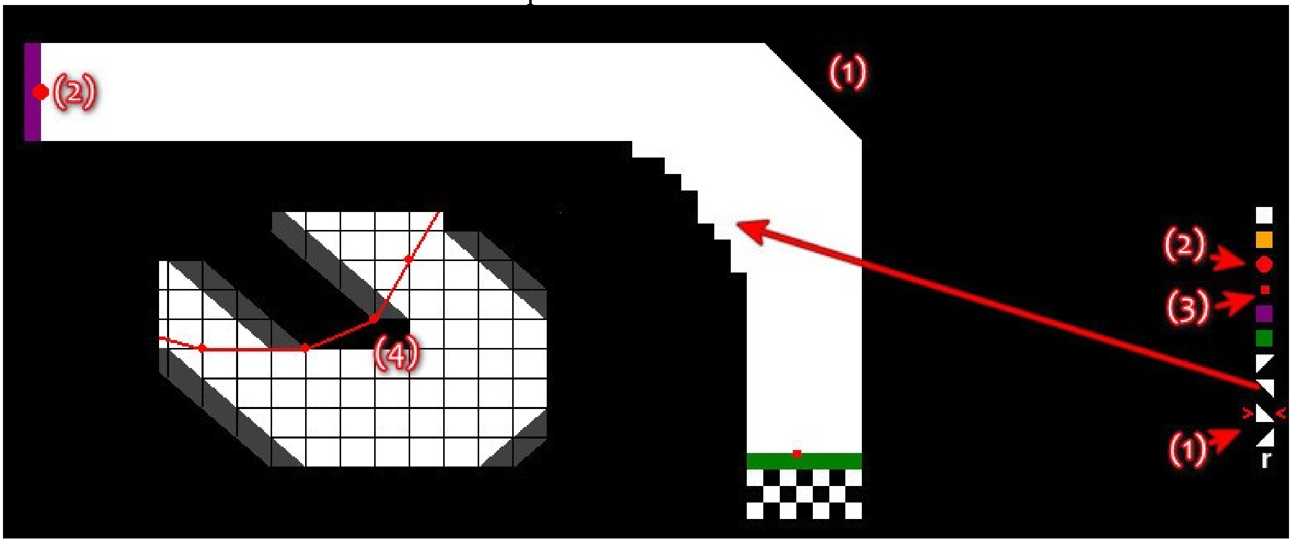
Choose New Track to start from scratch, or select an existing track to edit

New Track
Ara_Made
Basica
Basicabis
Candelabro
El_astronauta
El_astronauta_E
El_Candelabro
El_Pulpo
Espiral

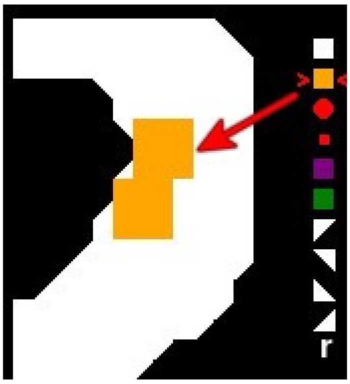
Once we make our choice, we will move to the design screen:



(A): button to return to the initial menu **(B)**: name of the track **(C)**: the UNDO, pbutton to undo the last actions, although its main use will be to reverse the fill tool (r) with which we can fill a dark area (framed by white lines) with white or vice versa a white area with black. The white track represents the track composed of white grids (default tool) and the "black" (right mouse button) is off the track. When we select a tool it will be framed between > <. The racing circuit must have a starting line: purple grids **(1)** and a finish line: green grids **(2)**. The starting line and the finish line can be horizontal or vertical but not oblique.

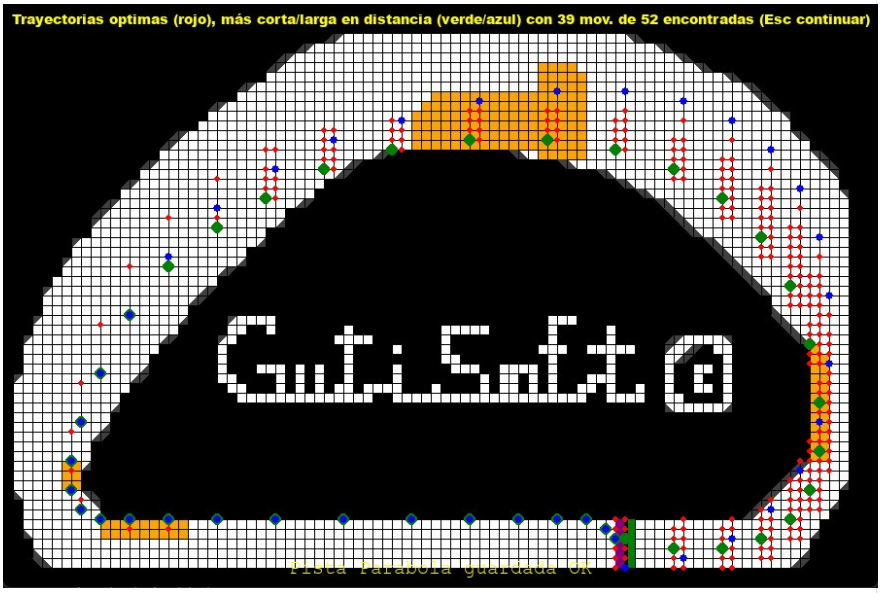


The other tools are: the "car" **(2)** a red circle that indicates the car's position on the starting line and must be attached to it, the "direction" **(3)** a small red square that indicates the direction the car must enter the finish line and must also be attached to it (both are necessary to save the track and if they are not present, we will be notified). The "triangles" **(1)** allow us to finish the corners creating shoulders that are considered part of the track and, to not be as fussy as the FIA, occasionally crossing a black square is also allowed (as long as the square is not surrounded by black squares) **(4)**. In the game, there will be no confusion because if we are out of the rules it will not let us make the movement.



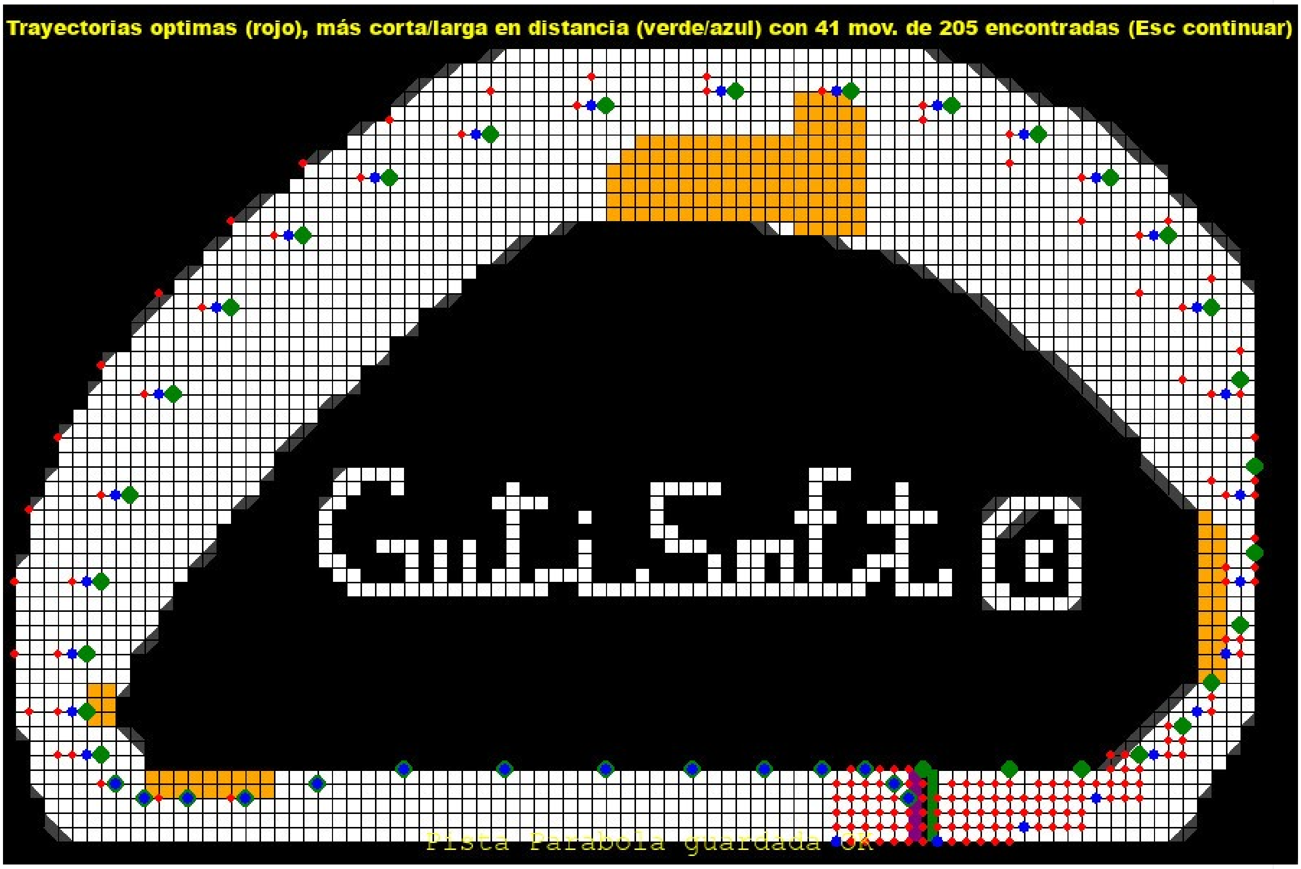
The last tool left to explain is the yellow grid (place obstacles) which has a very subtle way of changing the trajectories of the cars, let's see how: To design effective game tracks, especially if they are circular (continuous) where several laps can be done, we have a visual aid that is activated when we press SAVE TRACK, if we do it with the left mouse button, the optimal trajectories that the car can travel from the starting line and cross the finish line without taking into account the obstacles we have placed will be calculated and shown. All of them are shown in red, the longest in Euclidean terms in blue, and the shortest in green (sometimes they will coincide), and also the number of optimal trajectories found. If, on the other hand, we press SAVE TRACK with the right button, the trajectories will avoid the obstacles we have placed, being a way to make the cars follow different trajectories depending on whether the obstacles are activated or not, and although all are optimal (fewer movements), in these last ones the obstacles will make them have more movements and the car that follows them will take longer to reach the finish line. In both cases, the longest optimal trajectory will be saved in a file with the name of the track and the .trai extension, which will be followed by the purple car in the replays, and with the .troi extension, a trajectory (which is not shown) that is the shortest in Euclidean terms and will be followed by the green car in the replays. The last action performed (right or left button to save the track) will determine which optimal trajectory is saved (with or without obstacles). The game in DEMO mode activates the trajectories with obstacles in the car that goes first, making the cars overtake and compete against each other, which will make the replay of the game much more fun. The car that goes first always has an advantage, and this is a way to neutralize it (to achieve this effectively, the trajectories with obstacles should have at least 2 or 3 more movements than those without obstacles). These graphics will also help us in the design of tracks that have different ways to reach the finish line where we should try to make the trajectories distribute evenly by their length.

Obstacles do not affect the game in races, only in DEMO mode, but they can also help us design tracks that have several alternatives to reach the finish line, seeing how the optimal trajectories are distributed based on the path followed. Let's look at a couple of examples to make it clearer:In the image on the right, the optimal trajectories are seen without taking into account the obstacles, their distribution is shown, they have 39 movements and 52 have been found. In the image below, those that take into account the obstacles are shown and we see that more have been found: 205 and they have more movements, 41 three more than those that do not take into account the obstacles, therefore the car that follows the first ones will cut off the one that follows the second ones.



Trayectorias optimas (rojo), más corta/larga en distancia (verde/azul) con 39 mov. de 52 encontradas (Esc continuar)

Pista Parabola guardada OK

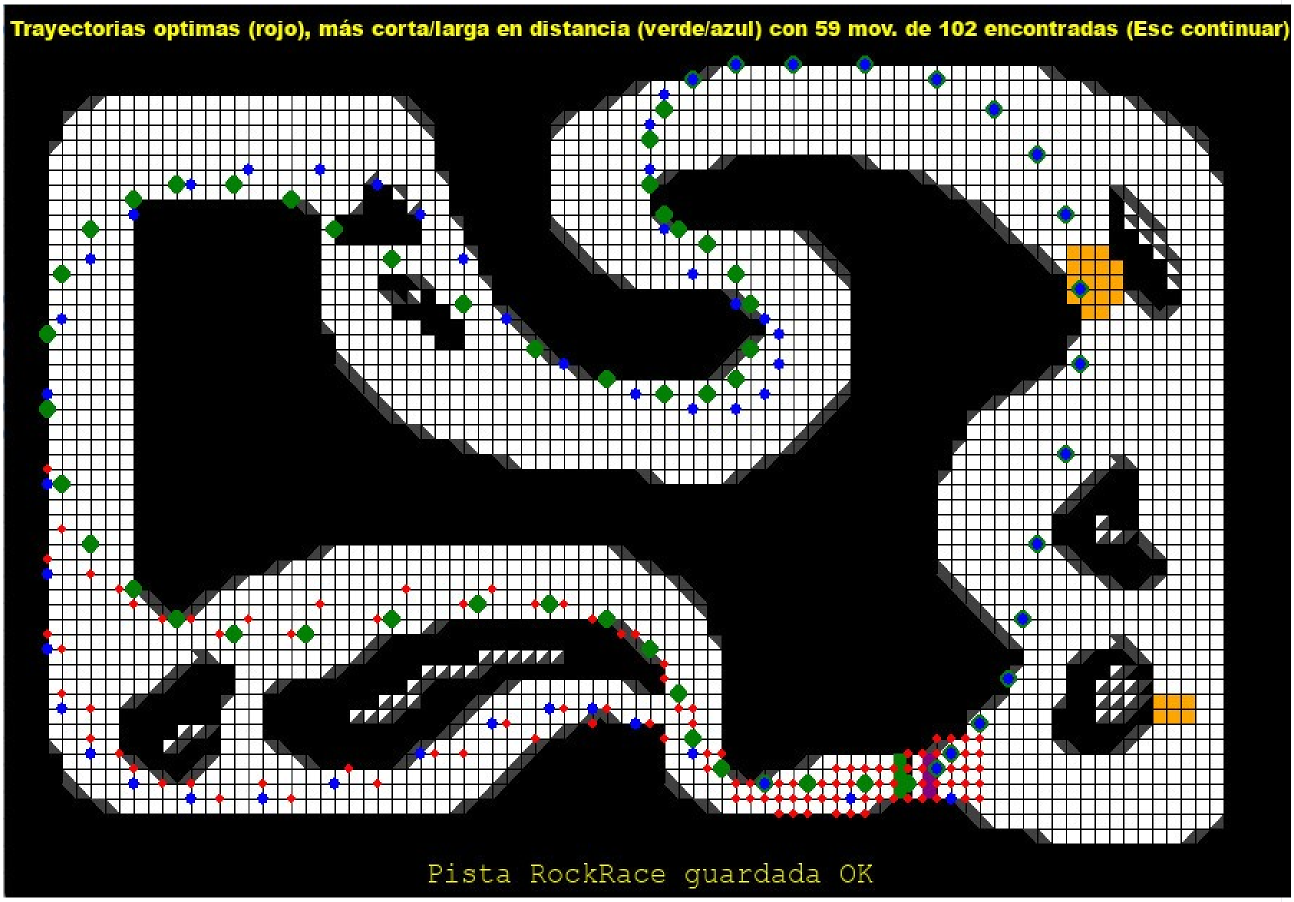


Trayectorias optimas (rojo), más corta/larga en distancia (verde/azul) con 41 mov. de 205 encontradas (Esc continuar)

Pista Parabola guardada OK

The trajectories we see start from the finish line with the cars stopped and will be the ones they follow in the first lap, but in the following laps when they pass the finish line, they already have speed and the following ones will be different, this is another thing to take into account when designing a circuit, especially when placing the finish line since if we put it very close to the first turn after the start the cars will not have time to brake and will crash, it is better to put the finish line near the entry curve.

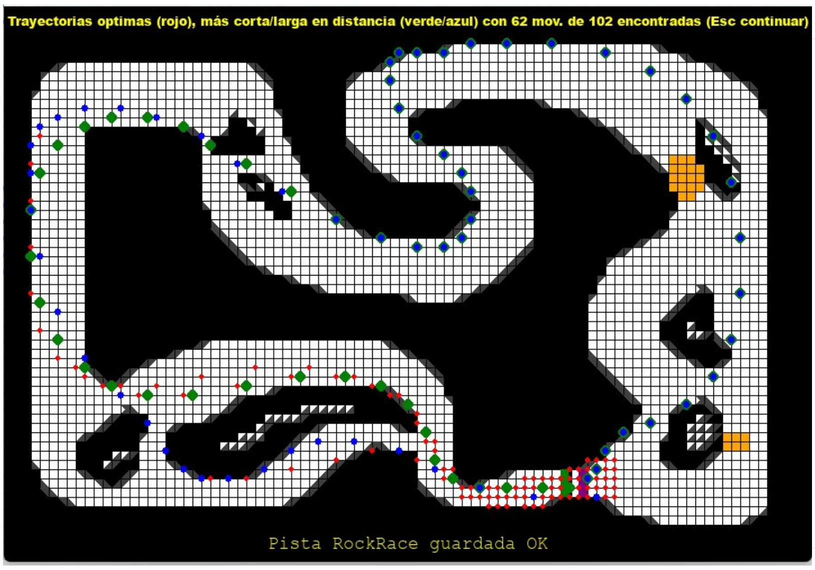
Let's see another example:



Trayectorias optimas (rojo), más corta/larga en distancia (verde/azul) con 59 mov. de 102 encontradas (Esc continuar)

Pista RockRace guardada OK

On this track that has several different paths to reach the finish line, we already see that the optimal trajectories are quite regularly distributed by their Euclidean length. In the upper image without considering the obstacles and in the lower one with:



Trayectorias optimas (rojo), más corta/larga en distancia (verde/azul) con 62 mov. de 102 encontradas (Esc continuar)

Pista RockRace guardada OK

If when designing a track we see that there are many optimal trajectories and that they are distributed in different paths, the track is well designed and will give "a lot of play", if on the contrary the trajectories overlap or there are very few, it is not a good track because any alternative to the finish line takes us out of "the optimum" and we should always follow the same line. For example, if we completely or partially block a path with an obstacle and we see that the movements of the optimal trajectories increase, this path is longer than the other alternatives, if on the contrary, they remain the same, we will know that it can also be chosen as an optimal path to reach the finish line.

We should also keep in mind that if we are designing a "non-circular" track we should leave some track after the finish line so that the cars can cross the finish line without going off the track and that they do not only have the option that the last movement is to touch it.

Another important detail is that if the track is circular (with continuity) in the first lap they start from stopped but in the following ones, they already start with speed so the finish line should not be very close to the first curve to avoid that when entering it with a lot of speed they crash. A good design is that the finish line (and the start) are closer to the last curve than to the first one.

Each time we are going to save an edited track an '_E' will be added to the name so as not to overwrite the original if by oversight we have not given it a new name (if we are doing tests it is enough to delete that addition so as not to lengthen the name and create multiple files).

© GutiSoft 2023