

# person



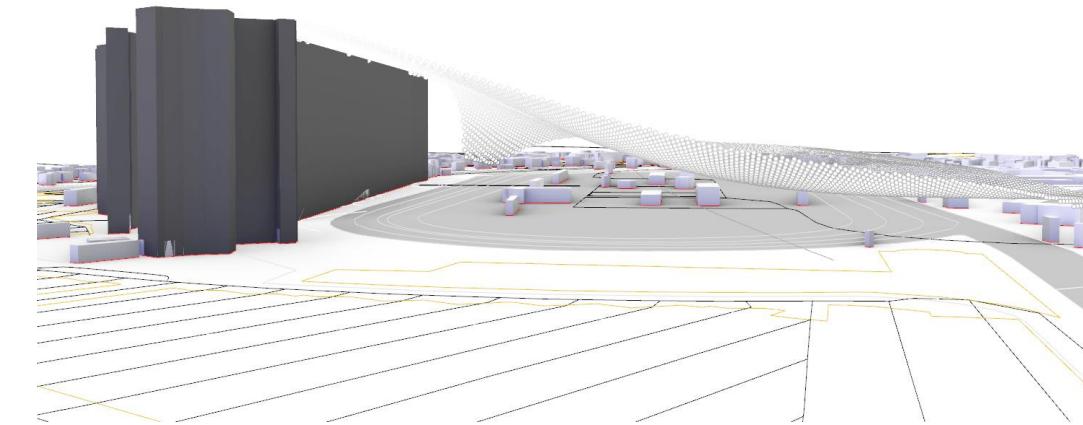
Diane Crump war die erste Frau, die als Jockey in einem offiziellen Pferderennen in den USA antrat. 1969 brach sie diese Geschlechterbarriere bei einem Rennen in Florida, trotz anfänglicher Proteste und Widerstände. Ein Jahr später schrieb sie erneut Geschichte, als sie die erste Frau wurde, die im prestigeträchtigen Kentucky Derby ritt. Ihr Mut und ihre Entschlossenheit machten sie zu einer Pionierin im Pferderennsport.

# code

```
project py [ ] finale_00.py [ ] hu_07B.py [ ] hu_07B.py [ ] D |  
479: # Step 3: Move additional vertices in their corresponding place  
480: if move_non_main_vertices:  
481:     vertex_quantity = len(vertices)  
482:     non_main_vertices_ids = [i for i in range(vertex_quantity) if i not in three_main_vertices_ids]  
483:     for i in non_main_vertices_ids:  
484:         vertex = vertices[i//vertex_quantity]  
485:         print vertex  
486:         projected_vertex = project_point_on_a_plane(normal, pi, vertex)  
487:         vertex_distance_to_plane = rs.VectorLength(rs.VectorSubtract(vertex, projected_vertex))  
488:         directional_vector = rs.VectorSubtract(vertex, projected_vertex)  
489:         print vertex  
490:         print "wdtp", vertex_distance_to_plane  
491:         if directional_vector != rs.CreateVector([0,0,0]):  
492:             # points_3d = bend_plane_circle(points_3d, plane_parameters, vertex[0:2], directional_vector, ver  
493:             previous_vertex = vertices[(i-1)//vertex_quantity]  
494:             next_vertex = vertices[(i+1)//vertex_quantity]  
495:             # Calcualte direction of a fold (bisector of the corner)  
496:             vector1 = rs.VectorSubtract(previous_vertex, vertex)  
497:             vector2 = rs.VectorSubtract(next_vertex, vertex)  
498:             parabola_direction = get_angle_bisector(vector1, vector2)  
499:             # Calculate height of the parabola  
500:             distance_between_neighbours= rs.Distance(previous_vertex, next_vertex)  
501:             corner_angle = rs.VectorAngle(vector1, vector2)  
502:             scale_factor = 1.25 # constant  
503:             parabola_height = scale_factor*distance_between_neighbours*(360-corner_angle)/360  
504:             # Calculate the focus distance  
505:             focus_distance = (parabola_height**2)/(2*distance_between_neighbours)  
506:             # Calculate the focus distance  
507:             focus_distance = (parabola_height**2)/(2*distance_between_neighbours)  
508:             # Calculate the focus distance  
509:             focus_distance = (parabola_height**2)/(2*distance_between_neighbours)  
510:             # Calculate the focus distance  
511:             focus_distance = (parabola_height**2)/(2*distance_between_neighbours)
```

Dieser Code berechnet geometrische Eigenschaften von Punkten in 3D, wie die Projektion auf eine Ebene, Abstände und Richtungen. Er ermittelt die Richtung einer Faltung (als Winkelhalbierende) und die Höhe einer Parabel basierend auf den Nachbarpunkten und ihrem Winkel. Diese Daten werden genutzt, um die Punkte entsprechend anzupassen oder zu verschieben.

# project



Das Dach über Racetrack