

## CNC (computer numerical control) milling machine simulator

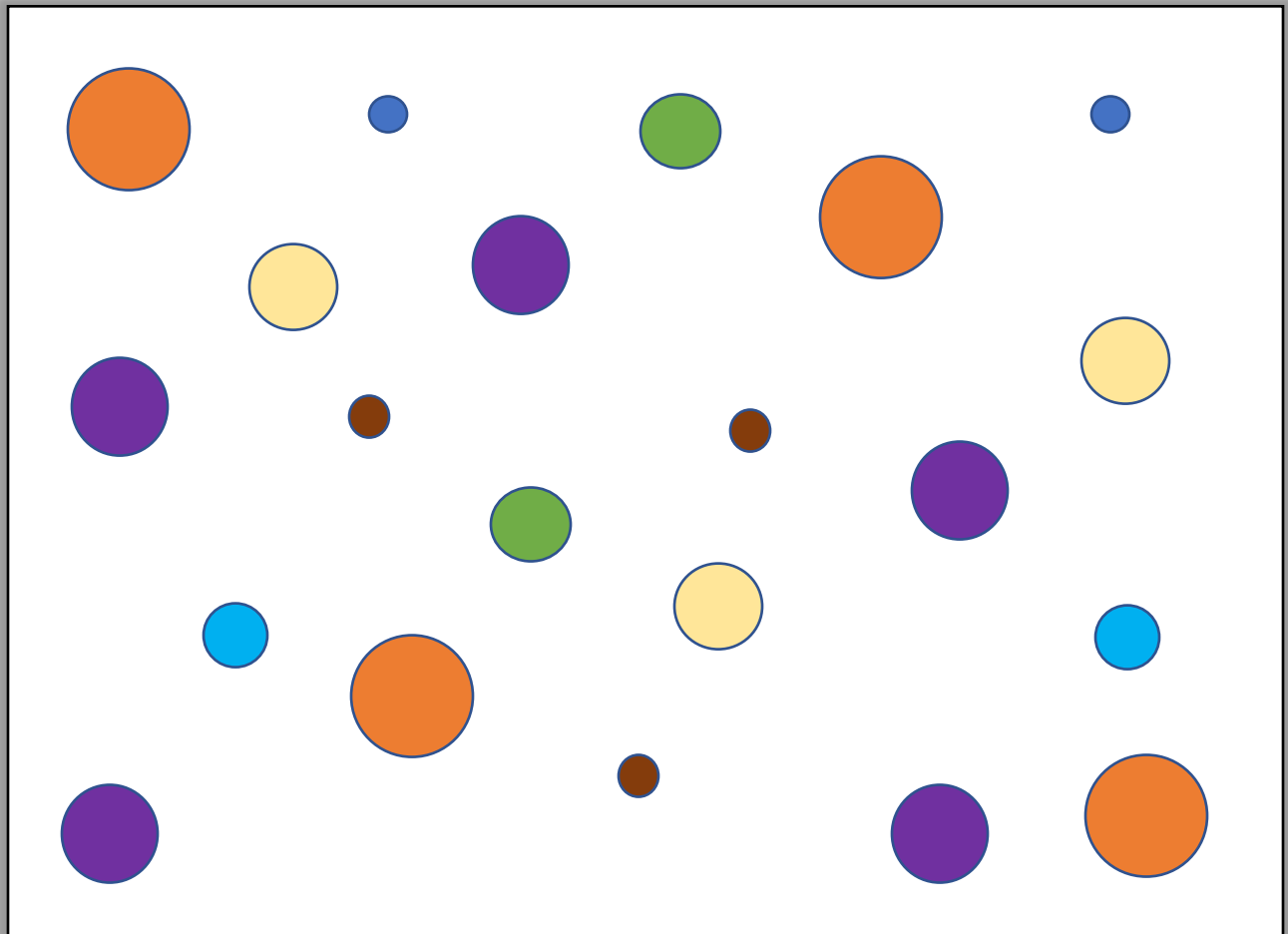
The following figure and table are a representation (an example) of what you need to develop. It does not mean your design must be exactly like this.

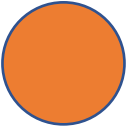







Your task is to ask the user how many shapes they want on this scene, or you can randomly generate those shapes on scene (Dynamically).

To make the above task easier for those who find it difficult to do it that way, you can have fixed points on space, and use that as your point (static).

Limit the 3D space so the objects are in the limited space for example: within 100 meters objects are distributed.

Each object has a Time to finish (TTF) which means every time you reach an object in space you have wait few seconds to finish that task (simulation of drilling/ molding machine)



Shapes								
TTF (Time to finish)	26	22	18	15	12	10	8	5

To find the shortest path to travels between nodes you can use the following algorithms:

- ❖ **Traveling Salesman Problem (TSP) algorithm**  
([https://en.wikipedia.org/wiki/Travelling\\_salesman\\_problem](https://en.wikipedia.org/wiki/Travelling_salesman_problem))
- ❖ **A\* algorithm**
- ❖ **Dijkstra's algorithm**
- ❖ **or Genetic algorithm (Optional)**

Example of these type of machines:

Machining of a plastic injection mold for a automotive exterior part

<https://www.youtube.com/watch?v=px77uA5tWCs&t=43s>

<https://www.youtube.com/watch?v=PlC2bHUBBns>

**Suggestion:** You can use this resource to implement Dijkstra's algorithm using the example they have used here too:

[https://en.wikipedia.org/wiki/Dijkstra%27s\\_algorithm](https://en.wikipedia.org/wiki/Dijkstra%27s_algorithm)

The point of this assignment is to learn how to use above algorithms implement them in games and other applications.

You have the flexibility to choose to make this dynamic or static for the creation of the nodes and connections. Some may find dynamic versions more difficult. Dynamic in this task means, the user can define the number of nodes and connections. This means it can be also randomized in a sense that nodes randomly populate the scene and connections are made randomly too.

**Suggestion:** You can create different scenes in your project and have a menu system that the user can choice with algorithm to run.

**Suggestion:** be flexible and creative to do this project, but you also can find a 2D solution and develop it in game engine and in 3D.

**Note 1:** Write a short report about how you managed to write the code, which sources you used. What type of help did you get? And a self-reflection about how hard you tried and how hard you worked on the task. Write how difficult and easy the task was for you, and which part was difficult. Write an estimated hour you spent working on this task and let me know if the time given to do this task was good enough.

**Note 2:** this is a group assignment.

Submission file format:

- Group: Group\_\_[number].[zip]
- Please note: [ ] are not to be included in the name.
- Grouping number links: [https://uinnlandet-my.sharepoint.com/:x:/g/personal/meisam\\_taheri\\_inn\\_no/ER0v-cmW-N9PgH3jEcNZ7TkB3rDiziOSMdN0Jec6cnaMrA?e=0WGzeO](https://uinnlandet-my.sharepoint.com/:x:/g/personal/meisam_taheri_inn_no/ER0v-cmW-N9PgH3jEcNZ7TkB3rDiziOSMdN0Jec6cnaMrA?e=0WGzeO)