# Robots

## Application in C# that simulates the process of producing elements and reports the progress to the UI.

To process a single element, paint it with three colors: blue, red, green. The elements are painted by robots. Each robot can only paint one element at the time. Painting each element takes a certain amount of time.

## Robot functions:

1. Red robot paints items with red paint.
2. Green robot paints items with green paint.
3. Blue robot paints items with blue paint.

## The process of making a finished element requires:

1. Item is initially located in a raw parts warehouse.
2. The item is transferred to a inactive robot. The robot paints the element in the appropriate color.
3. The element cannot be painted more than once with the same color.
4. If the element has not yet been painted with all three colors, it is returned to the raw parts warehouse.
5. The elements painted with all the required colors are transferred to the finished items warehouse.

## From the user’s point of view, the following operations are allowed:

1. Set the number of elements to be processed.
2. Set the number of robots painting in a given color.
3. Input the time required to finish the painting of a single element in the desired color.

## Implementation details

The implementation should be parallelized and optimized so the elements are processed as quickly as possible. It should be possible to interrupt the process at any time. After canceling, the only option is to restart the process. If not aborted, the process ends when all the items have been processed.

## The program graphical interface should contain the following:

1. Configuration section to input the process settings like number of robots and elements.
2. Statistics area with general process status, including:  
   2.1 The time elapsed since the process started.  
   2.2 The number and percentage of items completed.  
   2.3 The number and percentage of items that are yet to be processed.  
   2.4 The number and percentage of items painted by each of the three colors.
3. The table which shows all the elements being processed and their current status in real time. The table consists of the five columns:  
   3.1 An identifier given to the item when it was created.  
   3.2 Red, if an item has already been painted red, the cell contains the corresponding picture or value. If it hasn’t been painted, the cell is empty. If it is in the process of being painted there is a special indicator for this also.  
   3.3 Blue, the same as for the red column, the column corresponds to the blue color.  
   3.4 Green, the same as for the red column, except that the column is the green color.  
   3.5 Status, the current status of an item. Status can have one of the following values:
   * Idle, element is located in the raw parts warehouse.
   * Painted by Red, element is being painted by a red robot.
   * Painted by Blue, element is being painted by a red robot.
   * Painted by Green, element is being painted by a red robot.
   * Finished, job is done, item is transferred to the warehouse for finished items.
4. Robots section with information on the usability of robots:  
   4.1 Total number of robots of a given type.  
   4.2 How many robots of a particular type are currently painting the elements.  
   4.3 How many elements have already been painted.
5. Action buttons to start and stop the process.

## Sample application main screen

