Use Case 2: Remove Program Block

Main Success Scenario

1. The user moves the mouse cursor over a block in the ProgramArea, then presses the left mouse key, then moves the mouse cursor to the Palette, and then releases the left mouse key.
2. The system removes the selected block, as well as any blocks below within the same scope.

Extensions

1a. The user releases the mouse key, while the maximum number of blocks has already been reached. 1. The system removes the selected block; all blocks reappear again in the Palette.

Sequence diagram with lifelines: `:myCanvasWindow`, `:EventHandler`, `:AreaController`, `:Area`, `:ProgramArea`, `:Block`, `:Palette`

loop

`:myCanvasWindow` → `:EventHandler`: handleMouseEvent() [case dragged]

opt [getDraggedBlock() == null]

`:EventHandler` → `:Area`: block := getDraggedBlock(x, y, areaController.getProgramArea()))

`:Area` → `:ProgramArea`: startDragging()

`:ProgramArea` → `:ProgramArea`: removeBlock()

`:Palette` → `:Palette`: refreshBlocks()

`:EventHandler` → `:AreaController`: setDraggedBlock(block)

opt [getDraggedBlock() != null]

`:EventHandler` → `:Block`: setCoords()

`:myCanvasWindow` → `:EventHandler`: handleMouseEvent()

`:EventHandler` → `:AreaController`: handleMouseEvent() [case released]

opt [!getProgramArea().inRange(x, y)]

`:EventHandler` → `:AreaController`: setAvailableBlocks()