Advent of Code 2022 Day 14

Selected Fun Problems of the ACM Programming Contest

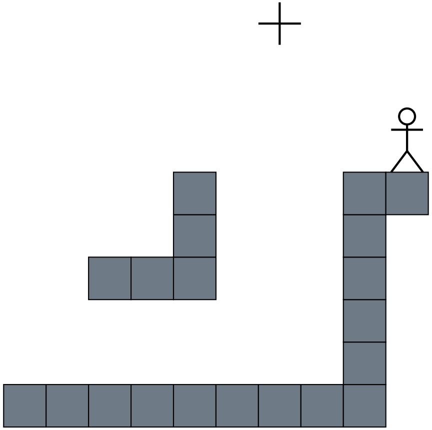
Simon Roller

University of Tübingen

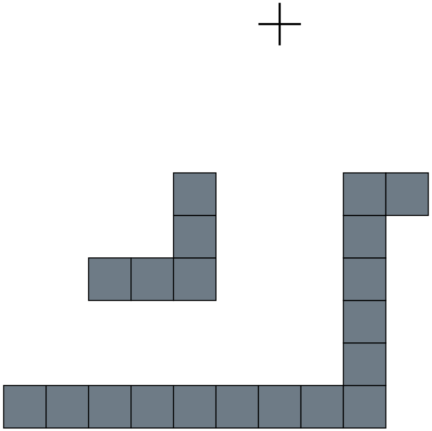
28.06.2024

Motivation

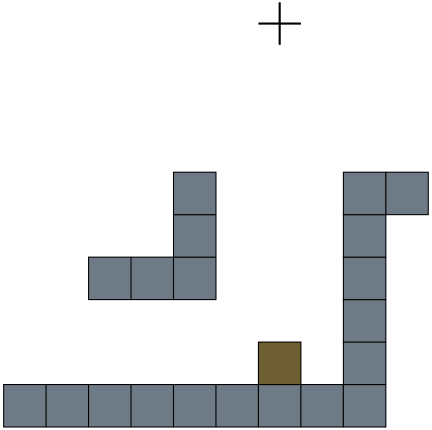
How much sand is needed to fill the cave and its surroundings?



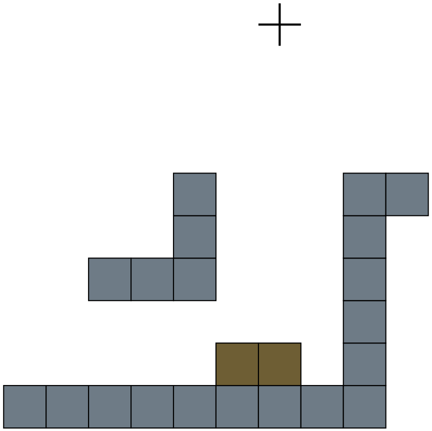
Problem Example



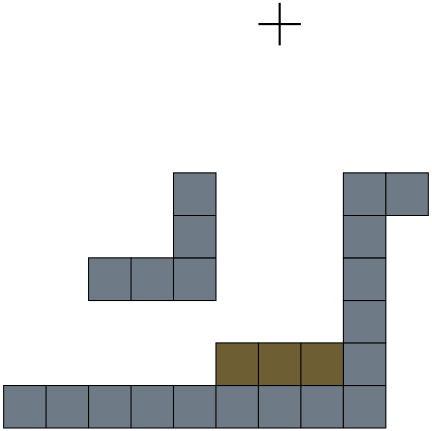
Problem Example



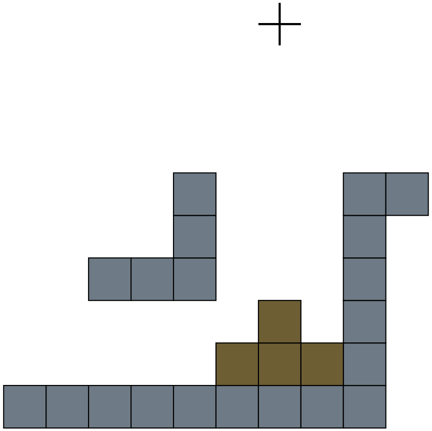
Problem Example



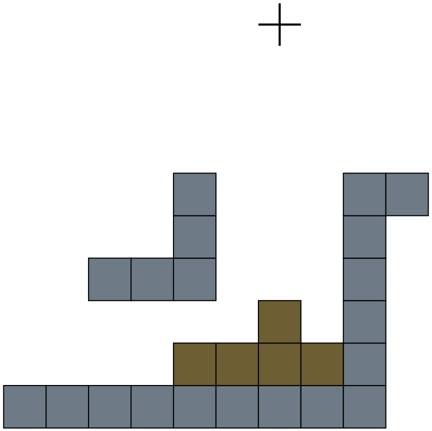
Problem Example



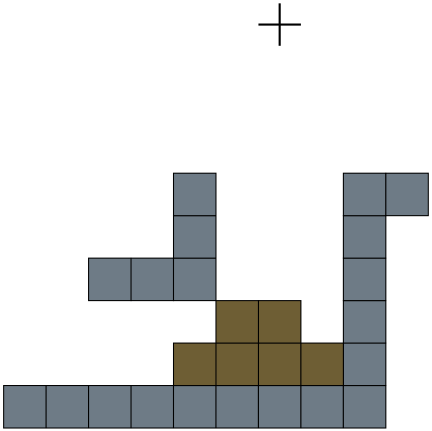
Problem Example



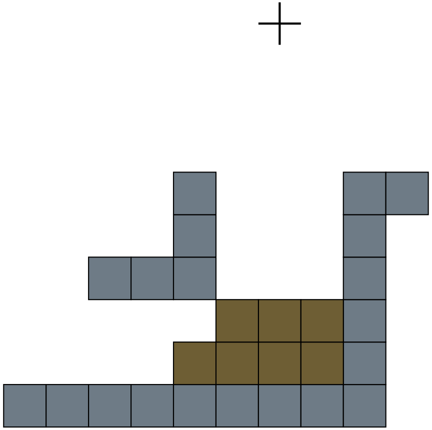
Problem Example



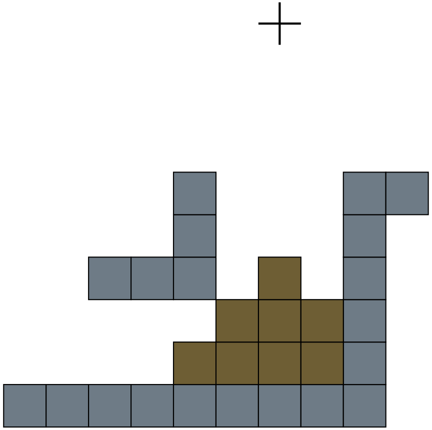
Problem Example



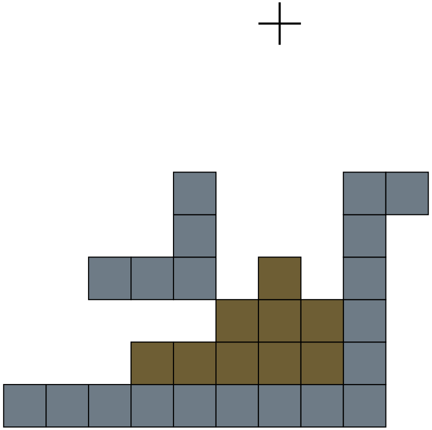
Problem Example



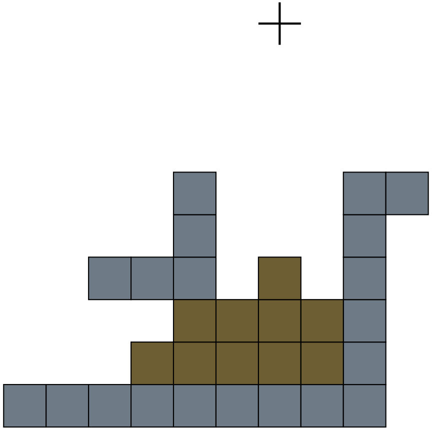
Problem Example



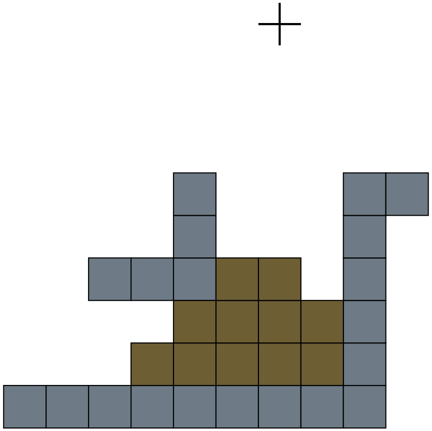
Problem Example



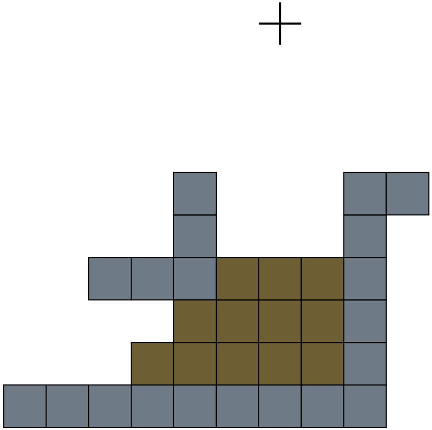
Problem Example



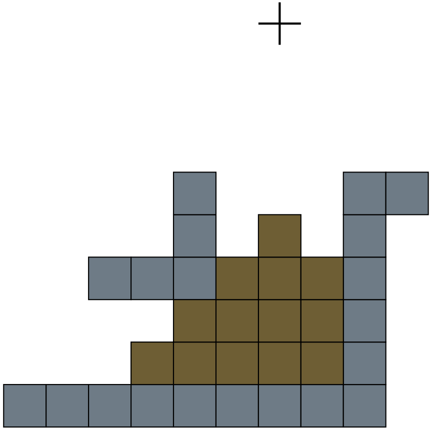
Problem Example



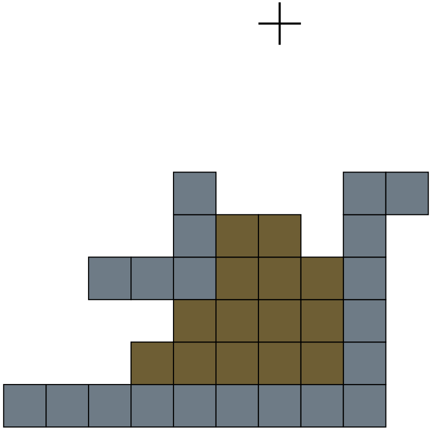
Problem Example



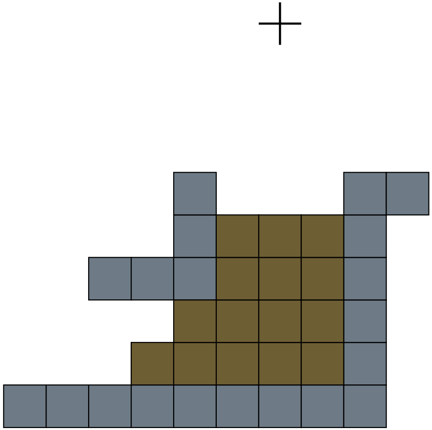
Problem Example



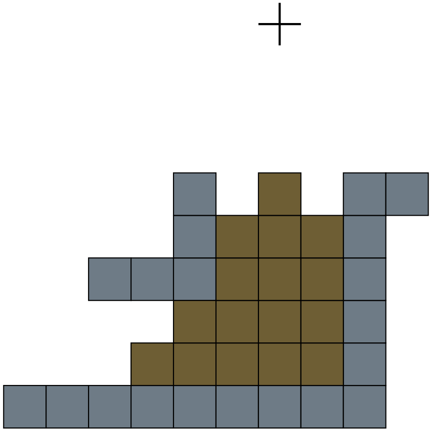
Problem Example



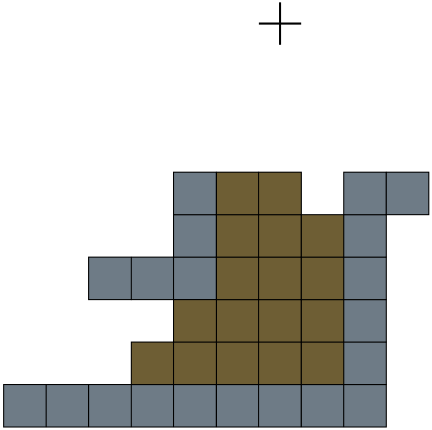
Problem Example



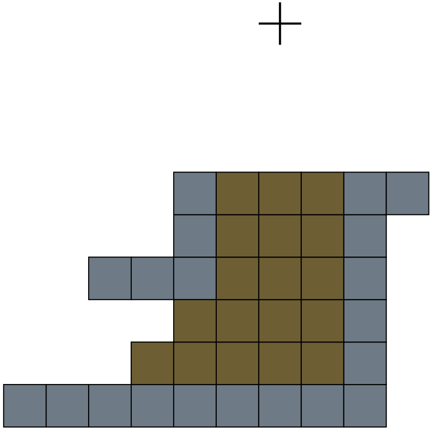
Problem Example



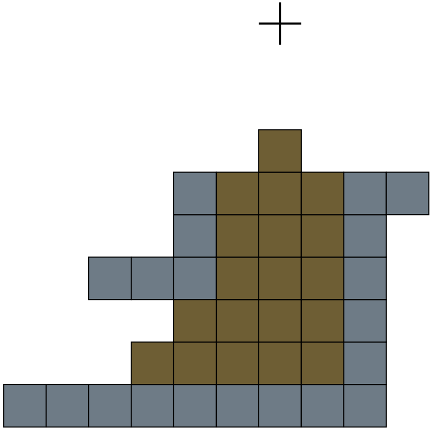
Problem Example



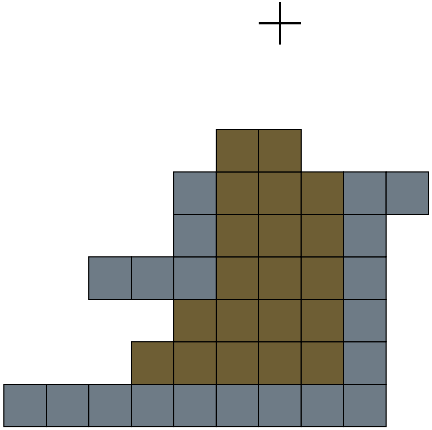
Problem Example



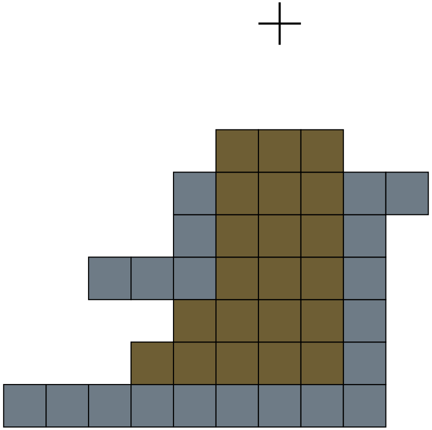
Problem Example



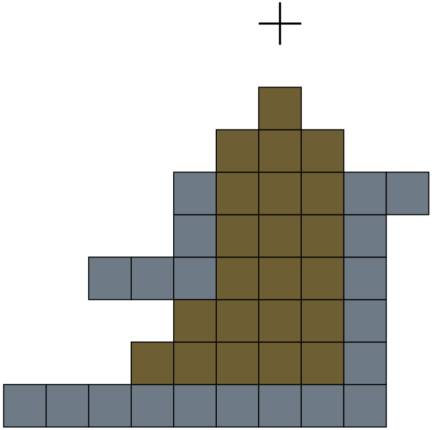
Problem Example



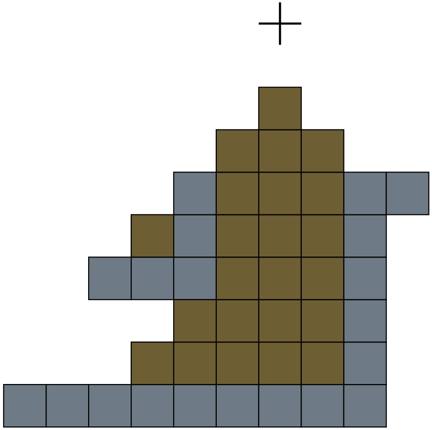
Problem Example



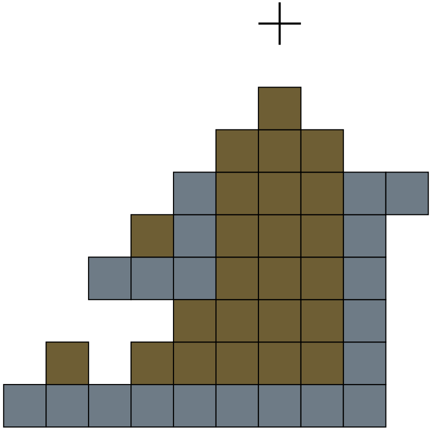
Problem Example



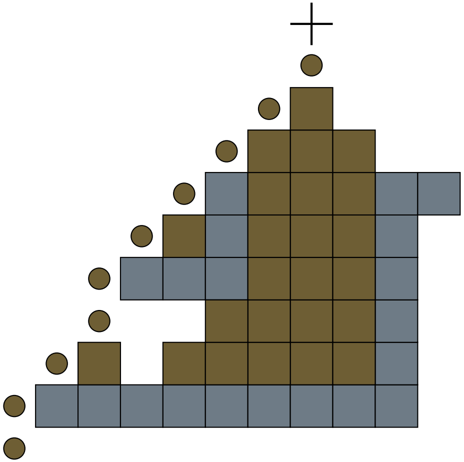
Problem Example



Problem Example



Problem Example



Programming Language

- ease of use

Programming Language

- ease of use
- no runtime or memory constraints

Programming Language

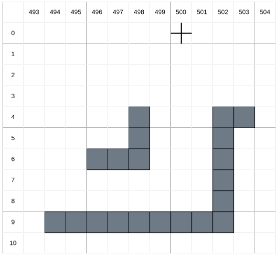
- ease of use
- no runtime or memory constraints
- me being proficient in the language

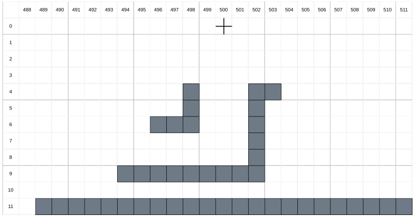
Input Details

```
498,4 -> 498,6 -> 496,6
503,4 -> 502,4 -> 502,9 -> 494,9
```

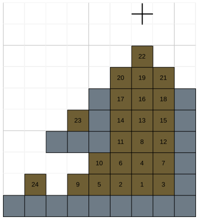
Input Details

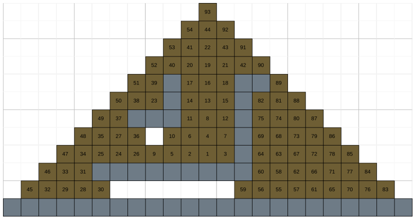
```
498,4 -> 498,6 -> 496,6
503,4 -> 502,4 -> 502,9 -> 494,9
```



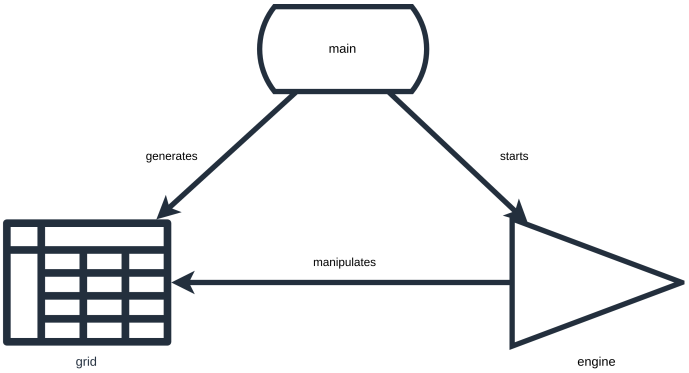


Output Details





Solution Approach



Code Example

```python
Coordinate = tuple[int, int]
Material = Enum('Material', 'air rock source solid_sand falling_sand')
Object = tuple[Material, Coordinate]
```

Code Example

```python
num = 0
while True:
  sand = (500, 0)
  while True:
```

Code Example

```python
num = 0
while True:
  sand = (500, 0)
  while True:
    if self.grid.is_air_at(Coordinate((sand[0], sand[1] + 1))):
      sand = (sand[0], sand[1] + 1)
```

Code Example

```python
num = 0
while True:
  sand = (500, 0)
  while True:
    if self.grid.is_air_at(Coordinate((sand[0], sand[1] + 1))):
      sand = (sand[0], sand[1] + 1)
    elif self.grid.is_air_at(Coordinate((sand[0] - 1, sand[1] + 1))):
      sand = (sand[0] - 1, sand[1] + 1)
```

Code Example

```python
num = 0
while True:
  sand = (500, 0)
  while True:
    if self.grid.is_air_at(Coordinate((sand[0], sand[1] + 1))):
      sand = (sand[0], sand[1] + 1)
    elif self.grid.is_air_at(Coordinate((sand[0] - 1, sand[1] + 1))):
      sand = (sand[0] - 1, sand[1] + 1)
    elif self.grid.is_air_at(Coordinate((sand[0] + 1, sand[1] + 1))):
      sand = (sand[0] + 1, sand[1] + 1)
```

Code Example

```python
num = 0
while True:
  sand = (500, 0)
  while True:
    if self.grid.is_air_at(Coordinate((sand[0], sand[1] + 1))):
      sand = (sand[0], sand[1] + 1)
    elif self.grid.is_air_at(Coordinate((sand[0] - 1, sand[1] + 1))):
      sand = (sand[0] - 1, sand[1] + 1)
    elif self.grid.is_air_at(Coordinate((sand[0] + 1, sand[1] + 1))):
      sand = (sand[0] + 1, sand[1] + 1)
    else:
      self.grid.add(Object((Material.solid_sand, Coordinate(sand))))
      break
```

Code Example

```python
num = 0
while True:
  sand = (500, 0)
  while True:
    if self.grid.is_air_at(Coordinate((sand[0], sand[1] + 1))):
      sand = (sand[0], sand[1] + 1)
    elif self.grid.is_air_at(Coordinate((sand[0] - 1, sand[1] + 1))):
      sand = (sand[0] - 1, sand[1] + 1)
    elif self.grid.is_air_at(Coordinate((sand[0] + 1, sand[1] + 1))):
      sand = (sand[0] + 1, sand[1] + 1)
    else:
      self.grid.add(Object((Material.solid_sand, Coordinate(sand))))
      break
    if sand[1] >= self.grid.get_last_row() + 1:
      if not part2:
        return num
      self.grid.add(Object((Material.solid_sand, Coordinate(sand))))
      break
```

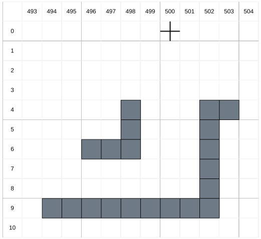
Code Example

```python
num = 0
while True:
  sand = (500, 0)
  while True:
    if self.grid.is_air_at(Coordinate((sand[0], sand[1] + 1))):
      sand = (sand[0], sand[1] + 1)
    elif self.grid.is_air_at(Coordinate((sand[0] - 1, sand[1] + 1))):
      sand = (sand[0] - 1, sand[1] + 1)
    elif self.grid.is_air_at(Coordinate((sand[0] + 1, sand[1] + 1))):
      sand = (sand[0] + 1, sand[1] + 1)
    else:
      self.grid.add(Object((Material.solid_sand, Coordinate(sand))))
      break
    if sand[1] >= self.grid.get_last_row() + 1:
      if not part2:
        return num
      self.grid.add(Object((Material.solid_sand, Coordinate(sand))))
      break
  num += 1
  if sand == (500, 0):
    return num
```

Live Demo

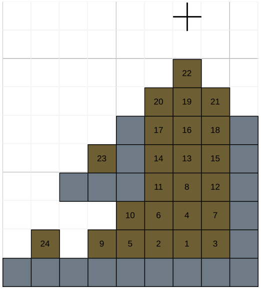
Key Takeaways & Outlook

- rock structure is created using the input coordinates



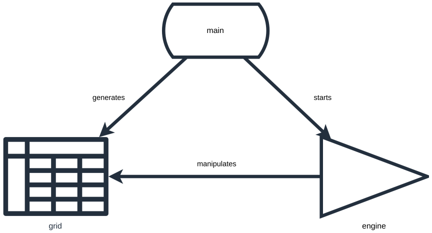
Key Takeaways & Outlook

- rock structure is created using the input coordinates
- dynamically calculate the number of sand



Key Takeaways & Outlook

- rock structure is created using the input coordinates
- dynamically calculate the number of sand
- adjust grid to be optimized for memory or computational performance



Key Takeaways & Outlook

- rock structure is created using the input coordinates
- dynamically calculate the number of sand
- adjust grid to be optimized for memory or computational performance
- render falling sand