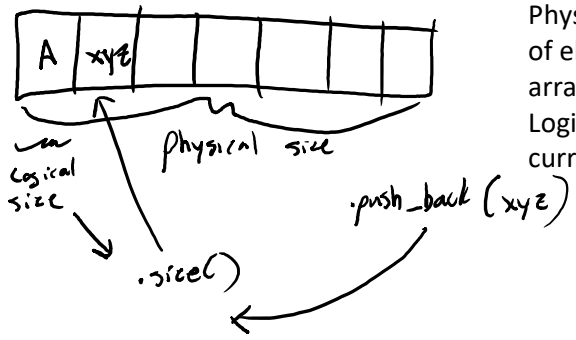
2019-09-10 Intro to Vectors

Tuesday, September 10, 2019 8:59 AM

- Vectors are dynamically expanding arrays
  - Unless we have a really good reason, we should prefer vectors over arrays in C++
- Basic initialization syntax
  - vector<DATA_TYPE> NAME_OF_VECTOR{};
  - e.g. vector<string> names{};
- Adding to vector
  - NAME_OF_VECTOR.push_back(ITEM)
  - e.g. names.push_back("Adam")
- Size of vector
  - NAME_OF_VECTOR.size()
  - e.g. names.size()

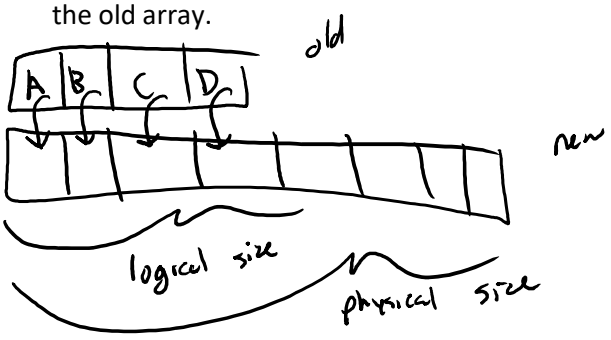
Adding items to a vector



Physical size: maximum number of elements that our underlying array can hold
Logical size: total number of elements currently in our array

What happens when you try to add to a vector whose physical size equals its logical size?

- In C++, we cannot extend the size of an existing array.
- Instead, we must dynamically allocate space for a new array size and then subsequently copy all elements from the old array.



- After copying the elements into the new array, we use delete[] to delete the old array's memory

- The cost of this operation is O(N)
    - If there are 4 items in old array, we make 4 copies
    - If there are 100 items in old array, we make 100 copies