☐ SABlythe / CSC24400-Spring2021

☆ 1 star **약** 0 forks ☆ Star Notifications Projects <> Code (!) Issues 11 Pull requests Actions Security ✓ Insights ې main ◄ CSC24400-Spring2021 / PFA_2_25 / PartiallyFilledArray.cpp SABlythe Added overloaded operator= to PFA class. (History A 1 contributor Raw **Blame** 85 lines (69 sloc) 1.51 KB #include "PartiallyFilledArray.hpp" 1 3 #include <iostream> 4 #include <string> 5 6 using namespace std; 7 8 PartiallyFilledArray::PartiallyFilledArray() 9 _array = new string[20]; 10 _currSize = 0; 11 12 13 PartiallyFilledArray(int maxSize) 14 15 16 _array = new string[maxSize]; _currSize = 0; 17 18 19 20 PartiallyFilledArray::add(const string &toAdd) 23 {

```
_array[_currSize] = toAdd;
24
25
       currSize++;
     }
26
27
     void
28
     PartiallyFilledArray::operator+=(const std::string &toAdd)
29
30
31
       add(toAdd);
     }
32
33
34
     void
     PartiallyFilledArray::operator+=(const PartiallyFilledArray &otherPFA)
       for (int index=0; index < otherPFA._currSize; index++)</pre>
37
38
39
            *this += otherPFA._array[index];
40
          }
     }
41
42
43
     string&
     PartiallyFilledArray::operator[] (int index)
44
45
46
       if (index < currSize)</pre>
47
         return _array[index];
       else
48
         {
49
            cerr << "invalid index (" << index << ") ... size is "</pre>
                 << currSize << endl;
51
            string *answer = new string;
            return *answer;
54
         }
     }
56
57
     const PartiallyFilledArray&
58
     PartiallyFilledArray::operator=(const PartiallyFilledArray &rhs)
59
       _array = new string[rhs._currSize];
61
       // fill it in
       for (int index=0; index< rhs. currSize; index++)</pre>
62
          _array[index] = rhs._array[index];
63
64
65
       currSize = rhs. currSize;
66
       return *this;
67
     }
68
69
     ostream&
70
71
     PartiallyFilledArray::print(ostream &toThisStream) const
```

```
72
73
       for (int index=0; index<_currSize; index++)</pre>
74
75
            toThisStream << *(_array+index) << endl;</pre>
76
          }
77
78
       return toThisStream;
     }
79
80
81
     std::ostream& operator<<(std::ostream &os, const PartiallyFilledArray &pfa)</pre>
82
83
       return pfa.print(os);
84
85
     }
```