**AP Classes: apvector**

|  |  |
| --- | --- |
| **Code** | **Effect** |
| apvector<char> a; | constructs a 0 element vector of char |
| apvector<int> a(10), b (20); | constructs a 10 element vector of int, indexed 0 to 9 |
| apvector<double> d(5, 0.0); | constructs a 5 element vector of double, initialized to 0.0 |
|  |  |
| b = a; | assignment works for apvectors (even of different sized vectors) |
| b.length(); | returns vector's current size (10) |
|  |  |
| d[0] = 7.5; | looks like an array |
| cout << d[2] << endl; | ditto |
|  |  |
| a.resize(50); | expandable arrays???     Indeed. |

**AP Classes: apmatrix**

|  |  |
| --- | --- |
| **Code** | **Effect** |
| apmatrix<char> a; | constructs a 0 element matrix |
| apmatrix<int> a(5, 10); | constructs a 5 row by 10 column matrix |
| apmatrix<int> b(5, 10, 0); | constructs a 5x10 matrix initialized to 0 |
|  |  |
| b = a; | assignment works |
| b.numrows(); | returns number of rows (5) |
| b.numcols(); | returns number of columns (10) |
|  |  |
| a[0][9] = 123; | assigns 123 to first row, tenth column |
| a[0,9] = 123; | illegal!!! |
|  |  |
| b.resize(10, 100); | expandable matrices, too! |