Course: Programming Fundamentals – **ENCM 339**

Lab #: Lab 8

Instructor: S. Norman

Student Name: **Mitchell Sawatzky**

Lab Section: **B02**

Date Submitted: **Nov 23, 2015**

# Exercise D

|  |
| --- |
| LabString& LabString::insert(size\_t pos, const char \*s)  {  int i = 0, j, k;  while (s[i] != '\0')  i++;  if (i == 0)  return \*this;  char\* new\_storage = new char[i + lengthM + 1];  for (j = 0, k = 0; j <= lengthM; j++) {  for (; j == pos && k < i; k++) {  new\_storage[j + k] = s[k];  }  if (storageM)  new\_storage[j + k] = storageM[j];  }  delete [ ] storageM;  storageM = new\_storage;  lengthM += i;  storageM[lengthM] = '\0';  return \*this;  } |

# Exercise E

|  |
| --- |
| LabVector::LabVector(const LabVector& src)  {  size\_t i, count = src.capacity();  storeM = new ElType[count];  end\_storeM = storeM + count;  for (i = 0; i < src.size(); i++)  storeM[i] = src.storeM[i];  end\_validM = storeM + i;  }  LabVector& LabVector::operator=(const LabVector& rhs)  {  if (this == &rhs)  return \*this;  delete [ ] storeM;  size\_t i, count = rhs.end\_storeM - rhs.storeM;  storeM = new ElType[count];  end\_storeM = storeM + count;  for (i = 0; i < rhs.end\_validM - rhs.storeM; i++)  storeM[i] = rhs.storeM[i];  end\_validM = &storeM[i];  return \*this;  }  void LabVector::resize(size\_t new\_size, const ElType& extra\_val)  {  if (new\_size == 0)  storeM = end\_validM = end\_storeM = 0;  else {  ElType\* new\_store = new ElType[new\_size];  size\_t count = size(), i;  for (i = 0; i < new\_size; i++) {  new\_store[i] = (i < count ? storeM[i] : extra\_val);  }  end\_storeM = end\_validM = new\_store + i;  delete [ ] storeM;  storeM = new\_store;  }  } |

|  |
| --- |
| Mitchell@ttys000 22:41 {0} [lab8]$ ./test.out  default constructor ...  EXPECT:  ACTUAL:  push\_back ...  EXPECT: -40 -30 -20 -10 0 10 20 30 40  ACTUAL: -40 -30 -20 -10 0 10 20 30 40  initialization with 4 copies of 65 ...  EXPECT: 65 65 65 65  ACTUAL: 65 65 65 65  initialization from built-in array ...  EXPECT: 0 1 4 9 16  ACTUAL: 0 1 4 9 16  copy constructor source, after updates ...  EXPECT: -88 -30 -20 -10 0 10 20 30 40 -99  ACTUAL: -88 -30 -20 -10 0 10 20 30 40 -99  copy constructor destination ...  EXPECT: -40 -30 -20 -10 0 10 20 30 40  ACTUAL: -40 -30 -20 -10 0 10 20 30 40  copy assignment operator, v4 after update ...  EXPECT: -77 -30 -20 -10 0 10 20 30 40  ACTUAL: -77 -30 -20 -10 0 10 20 30 40  copy assignment operator, v5 result ...  EXPECT: -40 -30 -20 -10 0 10 20 30 40  ACTUAL: -40 -30 -20 -10 0 10 20 30 40  copy assignment operator, v6 result ...  EXPECT: -40 -30 -20 -10 0 10 20 30 40  ACTUAL: -40 -30 -20 -10 0 10 20 30 40  resize ...  EXPECT: -88 -30 -20 -10  ACTUAL: -88 -30 -20 -10  EXPECT: -88 -30 -20 -10 0 0 0 0 0  ACTUAL: -88 -30 -20 -10 0 0 0 0 0  EXPECT: -88 -30 -20 -10 0 0 0 0 0 42 42  ACTUAL: -88 -30 -20 -10 0 0 0 0 0 42 42  EXPECT:  ACTUAL:  EXPECT: 99 99 99 99 99  ACTUAL: 99 99 99 99 99  EXPECT: 99 99 99 99 99 12 12 12 12 12 12  ACTUAL: 99 99 99 99 99 12 12 12 12 12 12 |