

Project: The Dynamic Array Class

For this project you are to implement a generic dynamic array class. It will be a templated class that provides the basic features of an array.

```
class RuntimeException // generic run-time exception
{
protected:
    std::string errorMsg;
public:
    RuntimeException(const std::string& err) { errorMsg = err; }
    std::string getMessage() const { return errorMsg; }
};

class InvalidIndex : public RuntimeException
{
public:
    InvalidIndex(const std::string& err): RuntimeException(err) {};
};

template <class dynElem>
class dynarr {
private:
    int capacity;
    dynElem *A;
public:
    dynarr(): capacity(0), A(0){};
    dynarr(int N): capacity(N), A(new dynElem[N]){}
    dynarr(const dynarr<dynElem> &other);
    ~dynarr();
    dynarr<dynElem> & operator=( const dynarr<dynElem> &other);
    dynElem & operator[](int ndx);
    int getCapacity();
    void reserve(int newcap);
    // if newcap <= capacity, does nothing;
    // if capacity is 0, allocates a dynamic array of
    // capacity newcap and makes A point to that array;
    // otherwise allocates a new dynamic array newA of capacity
    // newcap, copies values in A to newA, deletes A and sets
    // A equal to newA
};

// Provide the missing code for the class methods:
```

You will be given an incomplete file `dynarr.h` whose contents are shown above. You are to complete the code in that file and submit it. It is the only file you are to submit.

Also, you should create a program to test your implementation. Make sure your test file exercises all the methods of the `dynarray` class.