Exercise 6.2

```
#include <iostream> // Ex. 6.2
using namespace std;
class CAT
{
public:
CAT() {itsAge = 1;}
~CAT() { cout << "Destructor: itsAge = "<<itsAge<<"\n"; }
int GetAge() const {return itsAge;}
void SetAge(int age) {itsAge = age;}
private:
int itsAge;
};

int main()
{   CAT *Family = new CAT[10];
    for (int i=0; i<10; i++)
    {   Family[i].SetAge(2*i+1);
        cout <<' '<< Family[i].GetAge();
    }
    delete [] Family;  // Destructors are called for 10 times
        // Only Family[0] is deleted and run-time error if no [] is used
    return 0;
}
```

Ex. 6.2b

```
#include <iostream>
using namespace std;
class CAT
{
public:
CAT() {itsAge = 1;}
~CAT() { cout << "Destructor: itsAge = "<<itsAge<<"\n"; }
int GetAge() const {return itsAge;}
void SetAge(int age) {itsAge = age;}
private:
int itsAge;
};

int main()
{
    CAT *Family[10];  //Array of pointers
    int i;
    for (i=0; i<10; i++)
    {   Family[i] = new CAT;
        Family[i]->SetAge(2*i+1);
        cout <<' '<< Family[i]->GetAge();
    }
    for (i=9; i>=0; i--)
    {   delete Family[i];
    }
    return 0;
}
```

Ex. 6.2c

```
//Pointer of Array
#include <iostream>
```

```
#include <string.h>
using namespace std;
class ACCOUNT
{
public:
    void writename(char nm[]);
    char * readname();
private:
    char name[80];
};

void ACCOUNT::writename(char nm[])
{
    strncpy(name,nm,79);
}

char * ACCOUNT::readname()
{
    return name;
}

int main()
{
    ACCOUNT *User = new ACCOUNT[3];
    char name1[80], name2[80], name3[80];
    cout << "Please input 3 names:\n";
    cin >> name1;
    cin >> name2;
    cin >> name3;
    User[0].writename(name1);
    User[1].writename(name2);
    User[2].writename(name3);
    cout << "User[0].name: "<<User[0].readname() <<endl;
    cout << "User[1].name: "<<User[1].readname() <<endl;
    cout << "User[2].name: "<<User[2].readname() <<endl;
    delete [] User;
    return 0;
}
```