7.3 Defining a class

Private data members

In addition to public member functions, a class definition has ***private data members***: Variables that member functions can access but class users cannot. Private data members appear after the word "private:" in a class definition.

PARTICIPATION ACTIVITY 7.3.1: Private data members.

Start 2x speed

```
class Restaurant {                          // Keeps a user's review of a restaurant
   public:
      void SetName(string restaurantName);  // Sets the restaurant's name
      void SetRating(int userRating);       // Sets the rating (1-5, with 5 best)
      void Print();                         // Prints name and rating on one line
   private:
      string name;
      int rating;
};

int main() {
   Restaurant favLunchPlace;

   favLunchPlace.rating = 5;
   ...
}
```

Feedback?

PARTICIPATION ACTIVITY 7.3.2: Private data members.

Consider the example above.

1) After declaring Restaurant x, a class *user* can use a statement like x.name = "Sue's Diner".

   - True
   -

False

2) After declaring a Restaurant object x, a class user can use a statement like myString = x.name.

- True
- False

3) A class definition should provide comments along with each private data member so that a class user knows how those data members are used.

- True
- False

Feedback?

Defining a class' public member functions

A programmer defining a class first *declares* member functions after the word "public:" in the class definition. A ***function declaration*** provides the function's name, return type, and parameter types, but not the function's statements.

The programmer must also *define* each member function. A ***function definition*** provides a class name, return type, parameter names and types, and the function's statements. A member function definition has the class name and two colons (::), known as the ***scope resolution operator***, preceding the function's name. A member function definition can access private data members.

PARTICIPATION ACTIVITY 7.3.3: Defining a member function of a class using the scope resolution operator.

Start 2x speed

```
class MyClass {
  public:
     void Fct1();

  private:
     int numA;
};

void MyClass::Fct1() {
   numA = 0;
}
```

numA is visible to member functions

```
void Fct1() {
   numA = 0;
}
```

Wrong

Feedback?

Figure 7.3.1: A complete class definition, and use of that class.

```
#include <iostream>
#include <string>
using namespace std;

class Restaurant {                          // Info about a
restaurant
   public:
      void SetName(string restaurantName);  // Sets the
restaurant's name
      void SetRating(int userRating);       // Sets the rating (1-
5, with 5 best)
      void Print();                         // Prints name and
rating on one line

   private:
      string name;
      int rating;
};

// Sets the restaurant's name
void Restaurant::SetName(string restaurantName) {
   name = restaurantName;
}

// Sets the rating (1-5, with 5 best)
void Restaurant::SetRating(int userRating) {
   rating = userRating;
}

// Prints name and rating on one line
void Restaurant::Print() {
   cout << name << " -- " << rating << endl;
}

int main() {
   Restaurant favLunchPlace;
   Restaurant favDinnerPlace;

   favLunchPlace.SetName("Central Deli");
   favLunchPlace.SetRating(4);

   favDinnerPlace.SetName("Friends Cafe");
   favDinnerPlace.SetRating(5);

   cout << "My favorite restaurants: " << endl;
   favLunchPlace.Print();
   favDinnerPlace.Print();

   return 0;
}
```

```
My favorite
restaurants:
Central Deli -- 4
Friends Cafe -- 5
```

Feedback?

PARTICIPATION ACTIVITY 7.3.4: Class definition.

Consider the example above.

1) How is the Print() member function *declared*?

- `Print();`
- `void Print();`
- `void Restaurant::Print();`

2) How does the Print() member function's *definition* begin?

- `void Print();`
- `void Restaurant::Print();`
- `void Restaurant::Print()`

3) Could the Print() function's definition begin as follows?

```
void Restaurant::Print(int x)
```

- Yes
- No

4) Which private data members of class Restaurant do the Print() function definition's statements access?

- SetName
- favLunchPlace and favDinnerPlace
- name and rating

Feedback?

Example: RunnerInfo class

The RunnerInfo class below maintains information about a person who runs, allowing a class user to set the time run and the distance run, and to get the runner's speed. The subsequent question set asks for the missing parts to be completed.

Figure 7.3.2: Simple class example: RunnerInfo.

```
#include <iostream>
using namespace std;

class RunnerInfo {
   public:
      void SetTime(int timeRunSecs);       // Time run in seconds
      void SetDist(double distRunMiles);   // Distance run in miles
      double GetSpeedMph();                // Speed in miles/hour
   __(A)__
      int timeRun;
      double distRun;
};

void __(B)__::SetTime(int timeRunSecs) {
   timeRun = timeRunSecs;  // timeRun refers to data member
}

void __(C)__SetDist(double distRunMiles) {
   distRun = distRunMiles;
}

double RunnerInfo::GetSpeedMph() const {
   return distRun / (timeRun / 3600.0); // miles / (secs / (hrs / 3600 secs))
}

int main() {
   RunnerInfo runner1; // User-created object of class type RunnerInfo
   RunnerInfo runner2; // A second object

   runner1.SetTime(360);
   runner1.SetDist(1.2);

   runner2.SetTime(200);
   runner2.SetDist(0.5);

   cout << "Runner1's speed in MPH: " << runner1.__(D)__ << endl;
   cout << "Runner2's speed in MPH: " << __(E)__ << endl;

   return 0;
}
```

```
Runner1's speed in MPH: 12
Runner2's speed in MPH: 9
```

Feedback?

PARTICIPATION ACTIVITY 7.3.5: Class example: RunnerInfo.

Complete the missing parts of the figure above.

1) (A)

Check Show answer

Answer

private:

Data members are private and thus appear after the word "private:".

2) (B)

Check Show answer

Answer

RunnerInfo

The class name RunnerInfo (and ::) are prepended to the function name, to associate this function definition with the RunnerInfo class definition.

3) (C)

Check Show answer

Answer

RunnerInfo::

The class name RunnerInfo and :: are prepended to the function name, to associate this function definition with the RunnerInfo class definition.

4) (D)

Check Show answer

Answer

GetSpeedMph()

A class user can call an object's member function using the "." member access operator, followed by the member function's name (and any required arguments; no arguments are required here).

5) (E)

Check Show answer

Answer

runner2.GetSpeedMph()

The class user created two objects: runner1 and runner2. This statement calls a member function on the runner2 object.

Feedback?

Exploring further:

- Classes from cplusplus.com
- Classes from msdn.microsoft.com

CHALLENGE ACTIVITY 7.3.1: Classes.

Start

Type the program's output.

```
#include <iostream>
using namespace std;

class Person {
   public:
      void SetName(string nameToSet);
      string GetName() const;
   private:
      string name;
};

void Person::SetName(string nameToSet) {
   name = nameToSet;
}

string Person::GetName() const {
   return name;
}

int main() {
   string userName;
   Person person1;

   userName = "Sam";

   person1.SetName(userName);
   cout << "I am " << person1.GetName();

   return 0;
}
```

```
I am Sam
```

1 2 3

Check Next

Feedback?

CHALLENGE ACTIVITY 7.3.2: Basic class use.

Print person1's kids, apply the IncNumKids() function, and print again, outputting text as below. End each line with a newline.
Sample output for below program with input 3:

```
Kids: 3
New baby, kids now: 4
```

```
19 }
20
21 int PersonInfo::GetNumKids() const {
22    return numKids;
23 }
24
25 int main() {
26    PersonInfo person1;
27    int personsKids;
28
29    cin >> personsKids;
30
31    person1.SetNumKids(personsKids);
32
33    /* Your solution goes here  */
34    cout << "Kids: " << person1.GetNumKids() << endl;
35    person1.IncNumKids();
36    cout << "New baby, kids now: " << person1.GetNumKids() << endl;
37
38
39    return 0;
40 }
```

Run ✓ All tests passed

✓ Testing with input: 3

Your output

```
Kids: 3
New baby, kids now: 4
```

✓ Testing with input: 6

Your output

```
Kids: 6
New baby, kids now: 7
```

Feedback?

CHALLENGE ACTIVITY 7.3.3: Basic class definition.

Define the missing function. licenseNum is created as: (100000 * customID) + licenseYear, where customID is a function parameter. Sample output with inputs 2014 777:

```
Dog license: 77702014
```

```
   licenseNum = (100000 * customID) + licenseYear;
}

int DogLicense::GetLicenseNum() const {
   return licenseNum;
}

int main() {
   DogLicense dog1;
   int userYear;
   int userId;

   cin >> userYear;
   cin >> userId;

   dog1.SetYear(userYear);
   dog1.CreateLicenseNum(userId);
   cout << "Dog license: " << dog1.GetLicenseNum() << endl;

   return 0;
}
```

Run ✓ All tests passed

✓ Testing with inputs: 2014 777

Your output `Dog license: 77702014`

✓ Testing with inputs: 2009 321

Your output `Dog license: 32102009`

Feedback?