C++ Function Overloading

Catherine Gamboa

Why do function overloading?

How to do function overloading.

Code Demo

Why do function overloading?

You write a small program to help a teacher

A function for averaging two integers

```
1  int gradeAve(int grade1,int grade2){
2    int average = (grade1 + grade2)/2;
3    return average;
4  }
```

```
1  cout<<"Average = " << gradeAve(98, 45)<<"\n";
```

```
Average = 71
```

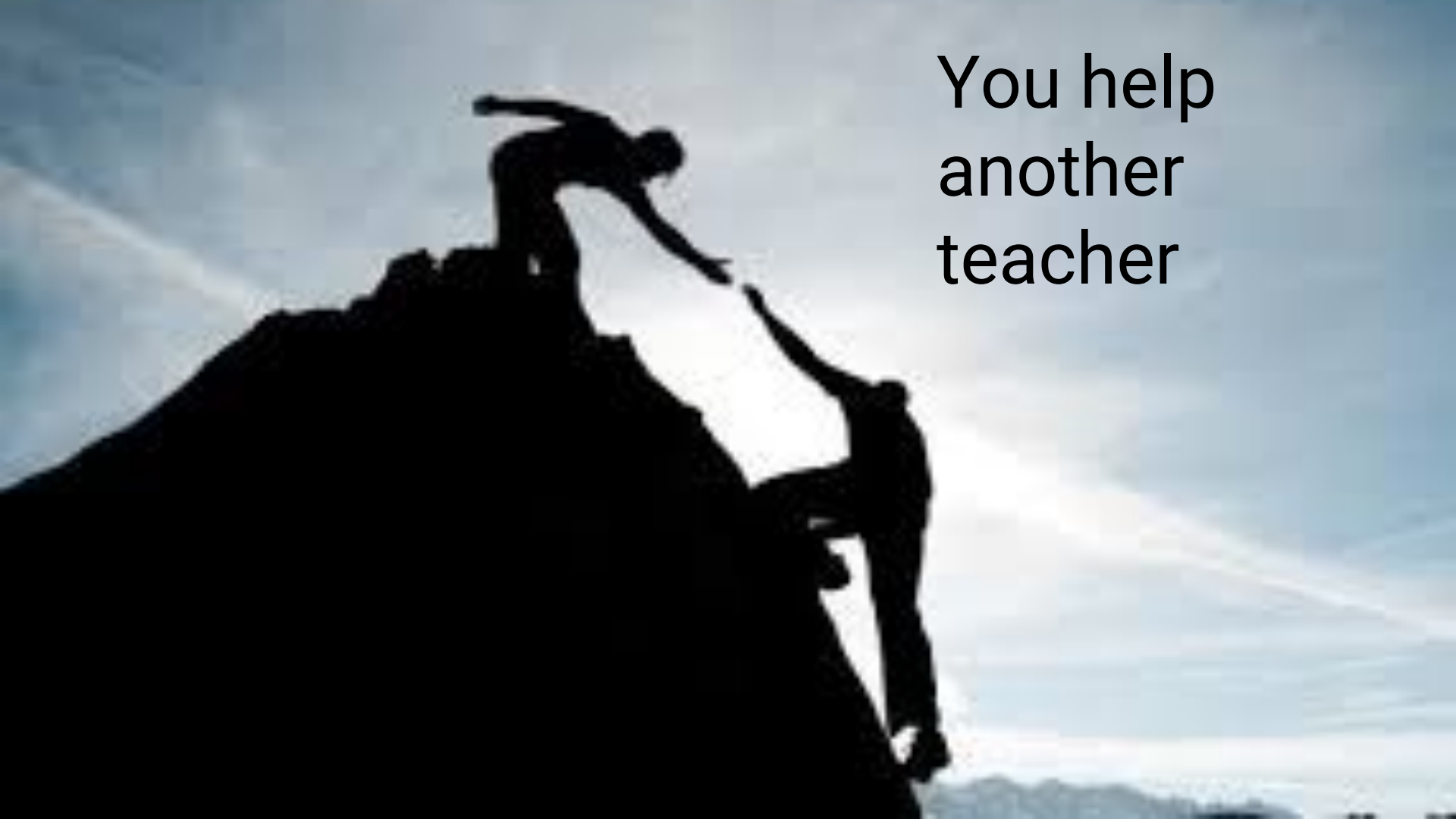
You help another teacher

A function for averaging two floats

```
1  float aveGradeFloat(float fgrade1,float fgrade2){
2    float average = (fgrade1 + fgrade2)/2;
3      std::cout<<average<<"\n";
4    return average;
5  }
```

```
1  cout<<"Average = " << aveGradeFloat(51.5, 40.8)<<"\n";
```

```
Average = 46.15
46.15
```


You help
another
teacher

A function for averaging two characters

```
char aveGradeChar(char cgrade1,char cgrade2){
    int g1 = convertGrade(cgrade1);
    int g2 = convertGrade(cgrade2);
    int average = (g1 + g2)/2;
    char ave = convertToLetter(average);
    return ave;
}
```

```
cout<<"Average = " << aveGradeChar('A','C')<<"\n";
```

```
Average = B
```


THE BETTER WAY
JUST AHEAD

Overloading

C++ allows you to specify more than one **definition** for a function name or an operator in the same scope, which is called function **overloading**


HOW?

1. Make a class for the functions.
2. Give the functions the same name.
3. Add the functions as public members of the class.
4. Define each function.

1 Make a class for the functions.

```
class ag{



};
```

2 Give the functions the same name.

```
class ag{

    int aveGrade(int igrade1,int igrade2);
    float aveGrade(float fgrade1,float fgrade2);
    char aveGrade(char cgrade1,char cgrade2);
};
```

3 Add the functions as public members of the class.

```
class ag{
public:
    int aveGrade(int igrade1,int igrade2);
    float aveGrade(float fgrade1,float fgrade2);
    char aveGrade(char cgrade1,char cgrade2);
};
```

4 Define each function

```
int ag::aveGrade(int igrade1,int igrade2){
   int average = (igrade1 + igrade2)/2;
   return average;
}
```

4 Define each function

```
float ag::aveGrade(float fgrade1,float fgrade2){
   float average = (fgrade1 + fgrade2)/2;
   return average;
};
```

4 Define each function

```
char ag::aveGrade(char cgrade1,char cgrade2){
    int g1 = 0;
    int g2 = 0;
    g1 = convertToInt(cgrade1);
    g2 = convertToInt(cgrade2);
    int average = (g1 + g2)/2;
    char ave = convertToChar(average);
    std::cout<<ave<<"\n";
  return ave;
}
```

Use the functions in a program

- Instantiate an instance of the class

```
1  ag a;
```

Use the functions in a program

- Instantiate an instance of the class
- Explicitly define the input variables

```
1  ag a;
2  int g1 = 98;
3  int g2 = 45;
4  float g3 = 51.5;
5  float g4 = 40.8;
6  char g5 = 'A';
7  char g6 = 'C';
8
```

Use the functions in a program

- Instantiate an instance of the class
- Explicitly define the input variables
- Use the function

```
1  ag a;
2  int g1 = 98;
3  int g2 = 45;
4  float g3 = 51.5;
5  float g4 = 40.8;
6  char g5 = 'A';
7  char g6 = 'C';
8
9  int gr1 = a.aveGrade(g1, g2);
10 float grf = a.aveGrade(g3, g4);
11 char grc = a.aveGrade(g5, g6);
12
```

Use the functions in a program

- Instantiate an instance of the class
- Explicitly define the input variables
- Use the function
- Check the outputs of each function

```
1  ag a;
2  int g1 = 98;
3  int g2 = 45;
4  float g3 = 51.5;
5  float g4 = 40.8;
6  char g5 = 'A';
7  char g6 = 'C';
8
9  int gr1 = a.aveGrade(g1, g2);
10 float grf = a.aveGrade(g3, g4);
11 char grc = a.aveGrade(g5, g6);
12
13 cout<<gr1<<"\n";
14 cout<<grf<<"\n";
15 cout<<grc<<"\n";
```

Use the functions in a program

- Instantiate an instance of the class
- Explicitly define the input variables
- Use the function
- Check the outputs of each function
- Outputs are correct

```
ag a;
int g1 = 98;
int g2 = 45;
float g3 = 51.5;
float g4 = 40.8;
char g5 = 'A';
char g6 = 'C';

int gr1 = a.aveGrade(g1, g2);
float grf = a.aveGrade(g3, g4);
char grc = a.aveGrade(g5, g6);

cout<<gr1<<"\n";
cout<<grf<<"\n";
cout<<grc<<"\n";
```

```
71
46.15
B
```

Jupyter notebook