

```
1. /*****
2.  * buggy-0.c
3.  *
4.  * David J. Malan
5.  * malan@harvard.edu
6.  *
7.  * Should print 10 asterisks but doesn't!
8.  *****/
9.
10. #include <stdio.h>
11.
12. int main(void)
13. {
14.     for (int i = 0; i <= 10; i++)
15.         printf("*");
16. }
```

```
1. /*****
2.  * buggy-1.c
3.  *
4.  * David J. Malan
5.  * malan@harvard.edu
6.  *
7.  * Should print 10 asterisks, one per line, but doesn't!
8.  *****/
9.
10. #include <stdio.h>
11.
12. int main(void)
13. {
14.     for (int i = 0; i <= 10; i++)
15.         printf("*");
16.         printf("\n");
17. }
```

```
1. /**
2.  * floats-0.c
3.  *
4.  * David J. Malan
5.  * malan@harvard.edu
6.  *
7.  * Tries to print 1/10 as a floating-point value.
8.  *
9.  * Demonstrates truncation.
10. */
11.
12. #include <stdio.h>
13.
14. int main(void)
15. {
16.     float f = 1 / 10;
17.     printf("%.1f\n", f);
18. }
```

```
1. /**
2.  * floats-1.c
3.  *
4.  * David J. Malan
5.  * malan@harvard.edu
6.  *
7.  * Prints 1/10 as a floating-point value to one decimal place.
8.  *
9.  * Demonstrates division of floating-point values.
10. */
11.
12. #include <stdio.h>
13.
14. int main(void)
15. {
16.     float f = 1.0 / 10.0;
17.     printf("%.1f\n", f);
18. }
```

```
1. /**
2.  * floats-2.c
3.  *
4.  * David J. Malan
5.  * malan@harvard.edu
6.  *
7.  * Prints 1/10 as a floating-point value to 28 decimal places.
8.  *
9.  * Demonstrates imprecision of floating-point values.
10. */
11.
12. #include <stdio.h>
13.
14. int main(void)
15. {
16.     float f = 1.0 / 10.0;
17.     printf("%.28f\n", f);
18. }
```

```
1. /**
2.  * function-0.c
3.  *
4.  * David J. Malan
5.  * malan@harvard.edu
6.  *
7.  * Prints a user's name.
8.  *
9.  * Demonstrates a function (not from a library) with a side effect.
10. */
11.
12. #include <cs50.h>
13. #include <stdio.h>
14.
15. // prototype
16. void PrintName(string name);
17.
18. int main(void)
19. {
20.     printf("Your name: ");
21.     string s = GetString();
22.     PrintName(s);
23. }
24.
25. /**
26.  * Says hello to someone by name.
27.  */
28. void PrintName(string name)
29. {
30.     printf("hello, %s\n", name);
31. }
```

```
1. /**
2.  * function-1.c
3.  *
4.  * David J. Malan
5.  * malan@harvard.edu
6.  *
7.  * Demands that user provide a positive integer.
8.  *
9.  * Demonstrates use of a function (not from a library) with a return value.
10. */
11.
12. #include <cs50.h>
13. #include <stdio.h>
14.
15. // prototype
16. int GetPositiveInt();
17.
18. int main(void)
19. {
20.     int n = GetPositiveInt();
21.     printf("Thanks for the %i!\n", n);
22. }
23.
24. /**
25.  * Gets a positive integer from a user.
26.  */
27. int GetPositiveInt(void)
28. {
29.     int n;
30.     do
31.     {
32.         printf("Please give me a positive int: ");
33.         n = GetInt();
34.     }
35.     while (n < 1);
36.     return n;
37. }
```

```
1. /*****
2.  * return.c
3.  *
4.  * David J. Malan
5.  * malan@harvard.edu
6.  *
7.  * Cubes a variable.
8.  *
9.  * Demonstrates use of parameter and return value.
10. *****/
11.
12. #include <stdio.h>
13.
14. // function prototype
15. int cube(int a);
16.
17. int main(void)
18. {
19.     int x = 2;
20.     printf("x is now %i\n", x);
21.     printf("Cubing...\n");
22.     x = cube(x);
23.     printf("Cubed!\n");
24.     printf("x is now %i\n", x);
25. }
26.
27. /**
28.  * Cubes argument.
29.  */
30. int cube(int n)
31. {
32.     return n * n * n;
33. }
```