

```
1:  /*****
2:   * adder.c
3:   *
4:   * Computer Science 50
5:   * David J. Malan
6:   *
7:   * Adds two numbers.
8:   *
9:   * Demonstrates use of CS 50's library.
10:  *****/
11:
12: #include <cs50.h>
13: #include <stdio.h>
14:
15: int
16: main(int argc, char * argv[])
17: {
18:     int x, y;
19:
20:     /* ask user for input */
21:     printf("Give me an integer: ");
22:     x = GetInt();
23:     printf("Give me another integer: ");
24:     y = GetInt();
25:
26:     /* do the math */
27:     printf("The sum of %d and %d is %d!\n", x, y, x + y);
28: }
```

```
1: /*****
2:  * conditions1.c
3:  *
4:  * Computer Science 50
5:  * David J. Malan
6:  *
7:  * Tells user if his or her input is positive or negative (somewhat
8:  * innacurately).
9:  *
10:  * Demonstrates use of if-else construct.
11: *****/
12:
13: #include <cs50.h>
14: #include <stdio.h>
15:
16: int
17: main(int argc, char * argv[])
18: {
19:     int n;
20:
21:     /* ask user for an integer */
22:     printf("I'd like an integer please: ");
23:     n = GetInt();
24:
25:     /* analyze user's input (somewhat inaccurately) */
26:     if (n > 0)
27:         printf("You picked a positive number!\n");
28:     else
29:         printf("You picked a negative number!\n");
30: }
```

```
1:  /*****
2:  * conditions2.c
3:  *
4:  * Computer Science 50
5:  * David J. Malan
6:  *
7:  * Tells user if his or her input is positive or negative.
8:  *
9:  * Demonstrates use of if-else if-else construct.
10: *****/
11:
12: #include <cs50.h>
13: #include <stdio.h>
14:
15: int
16: main(int argc, char * argv[])
17: {
18:     int n;
19:
20:     /* ask user for an integer */
21:     printf("I'd like an integer please: ");
22:     n = GetInt();
23:
24:     /* analyze user's input (somewhat inaccurately) */
25:     if (n > 0)
26:         printf("You picked a positive number!\n");
27:     else if (n == 0)
28:         printf("You picked zero!\n");
29:     else
30:         printf("You picked a negative number!\n");
31: }
```

```
1:  /*****
2:   * f2c.c
3:   *
4:   * Computer Science 50
5:   * David J. Malan
6:   *
7:   * Converts Fahrenheit to Celsius.
8:   *
9:   * Demonstrates casting from int to char.
10:  *****/
11:
12:
13: #include <cs50.h>
14: #include <stdio.h>
15:
16: int
17: main(int argc, char * argv[])
18: {
19:     float f, c;
20:
21:     /* ask user user for temperature in Fahrenheit */
22:     printf("Temperature in F: ");
23:     f = GetFloat();
24:
25:     /* convert F to C */
26:     c = 5 / 9. * (f - 32);
27:
28:     /* display result */
29:     printf("%.1f F = %.1f C\n", f, c);
30: }
```

```
1:  /*****
2:   * hello1.c
3:   *
4:   * Computer Science 50
5:   * David J. Malan
6:   *
7:   * Says hello to the world.
8:   *
9:   * Demonstrates use of printf.
10:  *****/
11:
12: #include <stdio.h>
13:
14: int
15: main(int argc, char * argv[])
16: {
17:     printf("hello, world\n");
18: }
```

```
1:  /*****
2:  * hello2.c
3:  *
4:  * Computer Science 50
5:  * David J. Malan
6:  *
7:  * Says hello to just David.
8:  *
9:  * Demonstrates use of CS 50's library.
10: *****/
11:
12: #include <cs50.h>
13: #include <stdio.h>
14:
15: int
16: main(int argc, char * argv[])
17: {
18:     string name = "David";
19:     printf("hello, %s\n", name);
20: }
```

```
1:  /*****
2:   * hello3.c
3:   *
4:   * Computer Science 50
5:   * David J. Malan
6:   *
7:   * Says hello to whomever.
8:   *
9:   * Demonstrates use of CS 50's library and standard input.
10:  *****/
11:
12: #include <cs50.h>
13: #include <stdio.h>
14:
15: int
16: main(int argc, char * argv[])
17: {
18:     string name;
19:     printf("State your name: ");
20:     name = GetString();
21:     printf("hello, %s\n", name);
22: }
```

```
1:  /*****
2:   * math1.c
3:   *
4:   * Computer Science 50
5:   * David J. Malan
6:   *
7:   * Computes a total but does nothing with it.
8:   *
9:   * Demonstrates use of variables.
10:  *****/
11:
12: #include <stdio.h>
13:
14: int
15: main(int argc, char * argv[])
16: {
17:     int x = 1;
18:     int y = 2;
19:     int z = x + y;
20: }
```



```
1:  /******
2:   * math2.c
3:   *
4:   * Computer Science 50
5:   * David J. Malan
6:   *
7:   * Computes and prints an integral total.
8:   *
9:   * Demonstrates use of a format string.
10:  *****/
11:
12: #include <stdio.h>
13:
14: int
15: main(int argc, char * argv[])
16: {
17:     int x = 1;
18:     int y = 2;
19:     int z = x + y;
20:     printf("%d", z);
21: }
```

```
1:  /*****
2:   * math3.c
3:   *
4:   * Computer Science 50
5:   * David J. Malan
6:   *
7:   * Computes and prints floating-point total.
8:   *
9:   * Demonstrates use of a format string with specified precision.
10:  *****/
11:
12: #include <stdio.h>
13:
14: int
15: main(int argc, char * argv[])
16: {
17:     float answer = 17 / 13.;
18:     printf("%.2f\n", answer);
19: }
```

```
1:  /*****
2:  * nonswitch.c
3:  *
4:  * Computer Science 50
5:  * David J. Malan
6:  *
7:  * Assesses the size of user's input.
8:  *
9:  * Demonstrates use of Boolean ANDing.
10: *****/
11:
12: #include <cs50.h>
13: #include <stdio.h>
14:
15: int
16: main(int argc, char * argv[])
17: {
18:     int n;
19:
20:     /* ask user for an integer */
21:     printf("Give me an integer between 1 and 10: ");
22:     n = GetInt();
23:
24:     /* judge user's input */
25:     if (n >= 1 && n <= 3)
26:         printf("You picked a small number.\n");
27:     else if (n >= 4 && n <= 6)
28:         printf("You picked a medium number.\n");
29:     else if (n >= 7 && n <= 10)
30:         printf("You picked a big number.\n");
31:     else
32:         printf("You picked an invalid number.\n");
33: }
```

```
1: /*****
2:  * positives.c
3:  *
4:  * Computer Science 50
5:  * David J. Malan
6:  *
7:  * Demands that user provide a positive number.
8:  *
9:  * Demonstrates use of do-while.
10: *****/
11:
12: #include <stdio.h>
13:
14: int
15: main(int argc, char * argv[])
16: {
17:     int n;
18:
19:     do
20:     {
21:         printf("I demand that you give me a positive integer: ");
22:         n = GetInt();
23:     }
24:     while (n < 1);
25:     printf("Thanks for the %d.\n", n);
26: }
```

```
1:  /*****
2:   * progress1.c
3:   *
4:   * Computer Science 50
5:   * David J. Malan
6:   *
7:   * Simulates a progress bar.
8:   *
9:   * Demonstrates sleep.
10:  *****/
11:
12: #include <stdio.h>
13: #include <unistd.h>
14:
15: int
16: main(int argc, char * argv[])
17: {
18:     int i;
19:
20:     /* simulate progress from 0% to 100% */
21:     for (i = 0; i <= 100; i++)
22:     {
23:         printf("Percent complete: %d%%\n", i);
24:         sleep(1);
25:     }
26:     printf("\n");
27: }
```

```
1:  /*****
2:   * progress2.c
3:   *
4:   * Computer Science 50
5:   * David J. Malan
6:   *
7:   * Simulates a better progress bar.
8:   *
9:   * Demonstrates \r, fflush, and sleep.
10:  *****/
11:
12: #include <stdio.h>
13: #include <unistd.h>
14:
15: int
16: main(int argc, char * argv[])
17: {
18:     int i;
19:
20:     /* simulate progress from 0% to 100% */
21:     for (i = 0; i <= 100; i++)
22:     {
23:         printf("\rPercent complete: %d%%", i);
24:         fflush(stdout);
25:         sleep(1);
26:     }
27:     printf("\n");
28: }
```

```
1:  /*****
2:   * progress3.c
3:   *
4:   * Computer Science 50
5:   * David J. Malan
6:   *
7:   * Simulates a better progress bar.
8:   *
9:   * Demonstrates a while loop.
10:  *****/
11:
12: #include <stdio.h>
13: #include <unistd.h>
14:
15: int
16: main(int argc, char * argv[])
17: {
18:     int i = 0;
19:
20:     /* simulate progress from 0% to 100% */
21:     while (i <= 100)
22:     {
23:         printf("\rPercent complete: %d%%", i);
24:         fflush(stdout);
25:         sleep(1);
26:         i++;
27:     }
28:     printf("\n");
29: }
```

```
1:  /******
2:   * sizeof.c
3:   *
4:   * Computer Science 50
5:   * David J. Malan
6:   *
7:   * Reports the sizes of C's data types.
8:   *
9:   * Demonstrates use of sizeof.
10:  *****/
11:
12: #include <stdio.h>
13:
14: int
15: main(int argc, char * argv[])
16: {
17:     /* local variables */
18:     char c;
19:     double d;
20:     float f;
21:     int i;
22:
23:     /* report the sizes of variables' types */
24:     printf("char: %d\n", sizeof(c));
25:     printf("double: %d\n", sizeof(d));
26:     printf("float: %d\n", sizeof(f));
27:     printf("int: %d\n", sizeof(i));
28: }
```



```
1:  /*****
2:  * switch1.c
3:  *
4:  * Computer Science 50
5:  * David J. Malan
6:  *
7:  * Assesses the size of user's input.
8:  *
9:  * Demonstrates use of a switch.
10: *****/
11:
12: #include <cs50.h>
13: #include <stdio.h>
14:
15: int
16: main(int argc, char * argv[])
17: {
18:     int n;
19:
20:     /* ask user for an integer */
21:     printf("Give me an integer between 1 and 10: ");
22:     n = GetInt();
23:
24:     /* judge user's input */
25:     switch (n)
26:     {
27:         case 1:
28:         case 2:
29:         case 3:
30:             printf("You picked a small number.\n");
31:             break;
32:
33:         case 4:
34:         case 5:
35:         case 6:
36:             printf("You picked a medium number.\n");
37:             break;
38:
39:         case 7:
40:         case 8:
41:         case 9:
42:         case 10:
43:             printf("You picked a big number.\n");
44:             break;
45:
46:         default:
47:             printf("You picked an invalid number.\n");
48:     }
49: }
```

```
1:  /*****
2:  * switch2.c
3:  *
4:  * Computer Science 50
5:  * David J. Malan
6:  *
7:  * Assesses a user's grade.
8:  *
9:  * Demonstrates use of a switch.
10: *****/
11:
12: #include <cs50.h>
13: #include <stdio.h>
14:
15: int
16: main(int argc, char * argv[])
17: {
18:     char c;
19:
20:     /* ask user for a char */
21:     printf("Pick a letter grade: ");
22:     c = GetChar();
23:
24:     /* judge user's input */
25:     switch (c)
26:     {
27:         case 'A':
28:         case 'a':
29:             printf("You picked an excellent grade.\n");
30:             break;
31:
32:         case 'B':
33:         case 'b':
34:             printf("You picked a good grade.\n");
35:             break;
36:
37:         case 'C':
38:         case 'c':
39:             printf("You picked a fair grade.\n");
40:             break;
41:
42:         case 'D':
43:         case 'd':
44:             printf("You picked a poor grade.\n");
45:             break;
46:
47:         case 'E':
48:         case 'e':
49:             printf("You picked a failing grade.\n");
50:             break;
51:
52:         default:
53:             printf("You picked an invalid grade.\n");
54:     }
55: }
```