

```
1  ../README.txt
2
3  hello{0,1}.c
4
5  buggy{0,1}.c
6
7  scores{0,1,2,3,4}.c
8
9  hi{0,1,2,3,4,5,6,7}.c
10
11 length{0,1,2}.c
12 string{0,1}.c
13 uppercase{0,1,2}.c
14
15 greet{0,1,2,3}.c
16
17 status.c
```

```
1  #include <stdio.h>
2
3  int main(void)
4  {
5      printf("hello, world\n");
6  }
```

```
1  #include <cs50.h>
2  #include <stdio.h>
3
4  int main(void)
5  {
6      string name = get_string("What's your name? ");
7      printf("hello, %s\n", name);
8  }
```

```
1 // Buggy example for printf
2
3 #include <stdio.h>
4
5 int main(void)
6 {
7     for (int i = 0; i <= 3; i++)
8     {
9         printf("#\n");
10    }
11 }
```

```
1 // Buggy example for debug50
2
3 #include <cs50.h>
4 #include <stdio.h>
5
6 int get_negative_int(void);
7
8 int main(void)
9 {
10     int i = get_negative_int();
11     printf("%i\n", i);
12 }
13
14 // Prompt user for positive integer
15 int get_negative_int(void)
16 {
17     int n;
18     do
19     {
20         n = get_int("Negative Integer: ");
21     }
22     while (n < 0);
23     return n;
24 }
```

```
1 // Averages three (hardcoded) numbers
2
3 #include <stdio.h>
4
5 int main(void)
6 {
7     // Scores
8     int score1 = 72;
9     int score2 = 73;
10    int score3 = 33;
11
12    // Print average
13    printf("Average: %f\n", (score1 + score2 + score3) / 3.0);
14 }
```

```
1 // Averages three (hardcoded) numbers using an array
2
3 #include <cs50.h>
4 #include <stdio.h>
5
6 int main(void)
7 {
8     // Scores
9     int scores[3];
10    scores[0] = 72;
11    scores[1] = 73;
12    scores[2] = 33;
13
14    // Print average
15    printf("Average: %f\n", (scores[0] + scores[1] + scores[2]) / 3.0);
16 }
```

```
1 // Averages three numbers using an array
2
3 #include <cs50.h>
4 #include <stdio.h>
5
6 int main(void)
7 {
8     // Get scores
9     int scores[3];
10    scores[0] = get_int("Score: ");
11    scores[1] = get_int("Score: ");
12    scores[2] = get_int("Score: ");
13
14    // Print average
15    printf("Average: %f\n", (scores[0] + scores[1] + scores[2]) / 3.0);
16 }
```

```
1 // Averages three numbers using an array and a loop
2
3 #include <cs50.h>
4 #include <stdio.h>
5
6 int main(void)
7 {
8     // Get scores
9     int scores[3];
10    for (int i = 0; i < 3; i++)
11    {
12        scores[i] = get_int("Score: ");
13    }
14
15    // Print average
16    printf("Average: %f\n", (scores[0] + scores[1] + scores[2]) / 3.0);
17 }
```

```
1 // Averages three numbers using an array, a constant, and a helper function
2
3 #include <cs50.h>
4 #include <stdio.h>
5
6 // Constant
7 const int N = 3;
8
9 // Prototype
10 float average(int length, int array[]);
11
12 int main(void)
13 {
14     // Get scores
15     int scores[N];
16     for (int i = 0; i < N; i++)
17     {
18         scores[i] = get_int("Score: ");
19     }
20
21     // Print average
22     printf("Average: %f\n", average(N, scores));
23 }
24
25 float average(int length, int array[])
26 {
27     // Calculate average
28     int sum = 0;
29     for (int i = 0; i < length; i++)
30     {
31         sum += array[i];
32     }
33     return sum / (float) length;
34 }
```

```
1 // Prints chars
2
3 #include <stdio.h>
4
5 int main(void)
6 {
7     char c1 = 'H';
8     char c2 = 'I';
9     char c3 = '!';
10
11     printf("%c%c%c\n", c1, c2, c3);
12 }
```

```
1 // Prints chars' ASCII codes
2
3 #include <stdio.h>
4
5 int main(void)
6 {
7     char c1 = 'H';
8     char c2 = 'I';
9     char c3 = '!';
10
11     printf("%i %i %i\n", c1, c2, c3);
12 }
```

```
1 // Prints string
2
3 #include <cs50.h>
4 #include <stdio.h>
5
6 int main(void)
7 {
8     string s = "HI!";
9     printf("%s\n", s);
10 }
```

```
1 // Prints string's ASCII codes
2
3 #include <cs50.h>
4 #include <stdio.h>
5
6 int main(void)
7 {
8     string s = "HI!";
9     printf("%i %i %i\n", s[0], s[1], s[2]);
10 }
```

```
1 // Prints string's ASCII codes, including NUL
2
3 #include <cs50.h>
4 #include <stdio.h>
5
6 int main(void)
7 {
8     string s = "HI!";
9     printf("%i %i %i %i\n", s[0], s[1], s[2], s[3]);
10 }
```

```
1 // Multiple strings
2
3 #include <cs50.h>
4 #include <stdio.h>
5
6 int main(void)
7 {
8     string s = "HI!";
9     string t = "BYE!";
10
11     printf("%s\n", s);
12     printf("%s\n", t);
13 }
```

```
1 // Array of strings
2
3 #include <cs50.h>
4 #include <stdio.h>
5
6 int main(void)
7 {
8     string words[2];
9
10    words[0] = "HI!";
11    words[1] = "BYE!";
12
13    printf("%s\n", words[0]);
14    printf("%s\n", words[1]);
15 }
```

```
1 #include <cs50.h>
2 #include <stdio.h>
3
4 int main(void)
5 {
6     string words[2];
7
8     words[0] = "HI!";
9     words[1] = "BYE!";
10
11     printf("%c%c%c\n", words[0][0], words[0][1], words[0][2]);
12     printf("%c%c%c%c\n", words[1][0], words[1][1], words[1][2], words[1][3]);
13 }
```

```
1 // Determines the length of a string
2
3 #include <cs50.h>
4 #include <stdio.h>
5
6 int main(void)
7 {
8     // Prompt for user's name
9     string name = get_string("Name: ");
10
11     // Count number of characters up until '\0' (aka NUL)
12     int n = 0;
13     while (name[n] != '\0')
14     {
15         n++;
16     }
17     printf("%i\n", n);
18 }
```

```
1 // Determines the length of a string using a function
2
3 #include <cs50.h>
4 #include <stdio.h>
5
6 int string_length(string s);
7
8 int main(void)
9 {
10     // Prompt for user's name
11     string name = get_string("Name: ");
12     int length = string_length(name);
13     printf("%i\n", length);
14 }
15
16 int string_length(string s)
17 {
18     // Count number of characters up until '\0' (aka NUL)
19     int n = 0;
20     while (s[n] != '\0')
21     {
22         n++;
23     }
24     return n;
25 }
```

```
1 // Determines the length of a string using a function
2
3 #include <cs50.h>
4 #include <stdio.h>
5 #include <string.h>
6
7 int main(void)
8 {
9     // Prompt for user's name
10    string name = get_string("Name: ");
11    int length = strlen(name);
12    printf("%i\n", length);
13 }
```

```
1 // Prints string char by char, using strlen
2
3 #include <cs50.h>
4 #include <stdio.h>
5 #include <string.h>
6
7 int main(void)
8 {
9     string s = get_string("Input: ");
10    printf("Output: ");
11    for (int i = 0; i < strlen(s); i++)
12    {
13        printf("%c", s[i]);
14    }
15    printf("\n");
16 }
```

```
1 // Prints string char by char, using strlen, remembering string's length
2
3 #include <cs50.h>
4 #include <stdio.h>
5 #include <string.h>
6
7 int main(void)
8 {
9     string s = get_string("Input: ");
10    printf("Output: ");
11    for (int i = 0, n = strlen(s); i < n; i++)
12    {
13        printf("%c", s[i]);
14    }
15    printf("\n");
16 }
```

```
1 // Uppercases a string
2
3 #include <cs50.h>
4 #include <stdio.h>
5 #include <string.h>
6
7 int main(void)
8 {
9     string s = get_string("Before: ");
10    printf("After: ");
11    for (int i = 0, n = strlen(s); i < n; i++)
12    {
13        if (s[i] >= 'a' && s[i] <= 'z')
14        {
15            printf("%c", s[i] - 32);
16        }
17        else
18        {
19            printf("%c", s[i]);
20        }
21    }
22    printf("\n");
23 }
```

```
1 // Uppercases string using ctype library (and an unnecessary condition)
2
3 #include <cs50.h>
4 #include <ctype.h>
5 #include <stdio.h>
6 #include <string.h>
7
8 int main(void)
9 {
10     string s = get_string("Before: ");
11     printf("After: ");
12     for (int i = 0, n = strlen(s); i < n; i++)
13     {
14         if (islower(s[i]))
15         {
16             printf("%c", toupper(s[i]));
17         }
18         else
19         {
20             printf("%c", s[i]);
21         }
22     }
23     printf("\n");
24 }
```

```
1 // Uppercases string using ctype library
2
3 #include <cs50.h>
4 #include <ctype.h>
5 #include <stdio.h>
6 #include <string.h>
7
8 int main(void)
9 {
10     string s = get_string("Before: ");
11     printf("After: ");
12     for (int i = 0, n = strlen(s); i < n; i++)
13     {
14         printf("%c", toupper(s[i]));
15     }
16     printf("\n");
17 }
```

```
1 // Uses get_string
2
3 #include <cs50.h>
4 #include <stdio.h>
5
6 int main(void)
7 {
8     string answer = get_string("What's your name? ");
9     printf("hello, %s\n", answer);
10 }
```

```
1 // Uses get_string with better-named variable
2
3 #include <cs50.h>
4 #include <stdio.h>
5
6 int main(void)
7 {
8     string name = get_string("What's your name? ");
9     printf("hello, %s\n", name);
10 }
```

```
1 // Prints a command-line argument
2
3 #include <cs50.h>
4 #include <stdio.h>
5
6 int main(int argc, string argv[])
7 {
8     if (argc == 2)
9     {
10         printf("hello, %s\n", argv[1]);
11     }
12     else
13     {
14         printf("hello, world\n");
15     }
16 }
```

```
1 // Prints command-line arguments
2
3 #include <cs50.h>
4 #include <stdio.h>
5
6 int main(int argc, string argv[])
7 {
8     for (int i = 0; i < argc; i++)
9     {
10         printf("%s\n", argv[i]);
11     }
12 }
```

```
1 // Returns explicit value from main
2
3 #include <cs50.h>
4 #include <stdio.h>
5
6 int main(int argc, string argv[])
7 {
8     if (argc != 2)
9     {
10         printf("Missing command-line argument\n");
11         return 1;
12     }
13     printf("hello, %s\n", argv[1]);
14     return 0;
15 }
```