

```
1  // Printing 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  // Printing 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  // Printing characters in an array of strings
2
3  #include <cs50.h>
4  #include <stdio.h>
5  #include <string.h>
6
7  int main(int argc, string argv[])
8  {
9      for (int i = 0; i < argc; i++)
10     {
11         for (int j = 0, n = strlen(argv[i]); j < n; j++)
12         {
13             printf("%c\n", argv[i][j]);
14         }
15         printf("\n");
16     }
17 }
```

---

```
1  // Explicitly casts chars to ints
2
3  #include <cs50.h>
4  #include <stdio.h>
5  #include <string.h>
6
7  int main(void)
8  {
9      string s = get_string("Name: ");
10     for (int i = 0; i < strlen(s); i++)
11     {
12         printf("%c %i\n", s[i], (int) s[i]);
13     }
14 }
```

---

```
1  // Buggy example for help50
2
3  int main(void)
4  {
5      printf("hello, world\n")
6  }
```

---

```
1 // Buggy example for help50
2
3 #include <stdio.h>
4
5 int main(void)
6 {
7     string s = get_string("Name: ");
8     printf("hello, %s\n", s);
9 }
```

---

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



```
1  // Capitalizes 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] - ('a' - 'A'));
16         }
17         else
18         {
19             printf("%c", s[i]);
20         }
21     }
22     printf("\n");
23 }
```

```
1  // Capitalizes 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  // Capitalizes 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  // 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 }
```

---

```
1 // Prints four question marks
2
3 #include <stdio.h>
4
5 int main(void)
6 {
7     printf("????\n");
8 }
```

---

```
1  // Prints four question marks using a loop
2
3  #include <stdio.h>
4
5  int main(void)
6  {
7      for (int i = 0; i < 4; i++)
8      {
9          printf("?");
10     }
11     printf("\n");
12 }
```

---

```
1 // Prints any number of question marks, as specified by user
2
3 #include <cs50.h>
4 #include <stdio.h>
5
6 int main(void)
7 {
8     int n = get_int("Number: ");
9     for (int i = 0; i < n; i++)
10     {
11         printf("?");
12     }
13     printf("\n");
14 }
```

```
1 // Prints a positive number of question marks, as specified by user
2
3 #include <cs50.h>
4 #include <stdio.h>
5
6 int main(void)
7 {
8     // Prompt user for a positive number
9     int n;
10    do
11    {
12        n = get_int("Positive number: ");
13    }
14    while (n <= 0);
15
16    // Print out that many bricks
17    for (int i = 0; i < n; i++)
18    {
19        printf("#\n");
20    }
21 }
```



```
1 // Prints a square of bricks, sized as specified by user
2
3 #include <cs50.h>
4 #include <stdio.h>
5
6 int main(void)
7 {
8     // Prompt user for a positive number
9     int n;
10    do
11    {
12        n = get_int("Positive number: ");
13    }
14    while (n <= 0);
15
16    // Print out this many rows
17    for (int i = 0; i < n; i++)
18    {
19        // Print out this many columns
20        for (int j = 0; j < n; j++)
21        {
22            printf("#");
23        }
24        printf("\n");
25    }
26 }
```

```
1 // Prints string char by char
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\n", s[i]);
14    }
15 }
```

```
1  // Prints string char by char, one per line
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:\n");
11     for (int i = 0, n = strlen(s); i < n; i++)
12     {
13         printf("%c\n", s[i]);
14     }
15 }
```

```
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 s = get_string("Name: ");
10
11     // Count number of characters up until '\0' (aka NUL)
12     int n = 0;
13     while (s[n] != '\0')
14     {
15         n++;
16     }
17     printf("%i\n", n);
18 }
```