```c
1   // Logical operators
2
3   #include <cs50.h>
4   #include <stdio.h>
5
6   int main(void)
7   {
8       // Prompt user to agree
9       char c = get_char("Do you agree?\n");
10
11      // Check whether agreed
12      if (c == 'Y' || c == 'y')
13      {
14          printf("Agreed.\n");
15      }
16      else if (c == 'N' || c == 'n')
17      {
18          printf("Not agreed.\n");
19      }
20  }
```

```c
 1   // Conditions and relational operators
 2
 3   #include <cs50.h>
 4   #include <stdio.h>
 5
 6   int main(void)
 7   {
 8       // Prompt user for x
 9       int x = get_int("x: ");
10
11       // Prompt user for y
12       int y = get_int("y: ");
13
14       // Compare x and y
15       if (x < y)
16       {
17           printf("x is less than y\n");
18       }
19       else if (x > y)
20       {
21           printf("x is greater than y\n");
22       }
23       else
24       {
25           printf("x is equal to y\n");
26       }
27   }
```

```c
// Opportunity for better design

#include <stdio.h>

int main(void)
{
    printf("cough\n");
    printf("cough\n");
    printf("cough\n");
}
```

```c
// Better design

#include <stdio.h>

int main(void)
{
    for (int i = 0; i < 3; i++)
    {
        printf("cough\n");
    }
}
```

```c
// Abstraction

#include <stdio.h>

void cough(void);

int main(void)
{
    for (int i = 0; i < 3; i++)
    {
        cough();
    }
}

// Cough once
void cough(void)
{
    printf("cough\n");
}
```

```c
// Abstraction with parameterization

#include <stdio.h>

void cough(int n);

int main(void)
{
    cough(3);
}

// Cough some number of times
void cough(int n)
{
    for (int i = 0; i < n; i++)
    {
        printf("cough\n");
    }
}
```

```c
 1    // Floating-point arithmetic with double
 2
 3    #include <cs50.h>
 4    #include <stdio.h>
 5
 6    int main(void)
 7    {
 8        // Prompt user for x
 9        double x = get_double("x: ");
10
11        // Prompt user for y
12        double y = get_double("y: ");
13
14        // Perform division
15        printf("x / y = %.50f\n", x / y);
16    }
```

```
1    // get_float and printf with %f
2
3    #include <cs50.h>
4    #include <stdio.h>
5
6    int main(void)
7    {
8        int price = get_float("What's the price?\n$");
9        printf("Your total is $%.2f.\n", price * 1.0625);
10   }
```

```c
1    // Floating-point arithmetic with float
2
3    #include <cs50.h>
4    #include <stdio.h>
5
6    int main(void)
7    {
8        // Prompt user for x
9        float x = get_float("x: ");
10
11        // Prompt user for y
12        float y = get_float("y: ");
13
14        // Perform division
15        printf("x / y = %.50f\n", x / y);
16   }
```

```
1   // A program that says hello to the world
2
3   #include <stdio.h>
4
5   int main(void)
6   {
7       printf("hello, world\n");
8   }
```

```
 1    // get_int and printf with %i
 2
 3    #include <cs50.h>
 4    #include <stdio.h>
 5
 6    int main(void)
 7    {
 8        int age = get_int("What's your age?\n");
 9        printf("You are at least %i days old.\n", age * 365);
10    }
```

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

```c
// Prints a row of 4 question marks with a loop

#include <stdio.h>

int main(void)
{
    for (int i = 0; i < 4; i++)
    {
        printf("?");
    }
    printf("\n");
}
```

```
1    // Prints a row of n question marks with a loop
2
3    #include <cs50.h>
4    #include <stdio.h>
5
6    int main(void)
7    {
8        int n;
9        do
10       {
11           n = get_int("Width: ");
12       }
13       while (n < 1);
14       for (int i = 0; i < n; i++)
15       {
16           printf("?");
17       }
18       printf("\n");
19   }
```

```
1    // Prints a column of 3 bricks
2
3    #include <stdio.h>
4
5    int main(void)
6    {
7        printf("#\n");
8        printf("#\n");
9        printf("#\n");
10   }
```

```
1    // Prints a column of 3 bricks with a loop
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   }
```

```c
 1    // Prints a column of n bricks with a loop
 2
 3    #include <cs50.h>
 4    #include <stdio.h>
 5
 6    int main(void)
 7    {
 8        int n;
 9        do
10        {
11            n = get_int("Height: ");
12        }
13        while (n < 1);
14        for (int i = 0; i < n; i++)
15        {
16            printf("#\n");
17        }
18    }
```

```
1    // Prints 3-by-3 grid of bricks
2
3    #include <stdio.h>
4
5    int main(void)
6    {
7        printf("###\n");
8        printf("###\n");
9        printf("###\n");
10   }
```

```
 1    // Prints a 3-by-3 grid of bricks with a loop
 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    // Prints an n-by-n grid of bricks with a loop
2
3    #include <cs50.h>
4    #include <stdio.h>
5
6    int main(void)
7    {
8        int n;
9        do
10       {
11           n = get_int("Size: ");
12       }
13       while (n < 1);
14       for (int i = 0; i < n; i++)
15       {
16           for (int j = 0; j < n; j++)
17           {
18               printf("#");
19           }
20           printf("\n");
21       }
22   }
```

```c
// Integer overflow

#include <stdio.h>
#include <unistd.h>

int main(void)
{
    // Iteratively double i
    for (int i = 1; ; i *= 2)
    {
        printf("%i\n", i);
        sleep(1);
    }
}
```

```c
1    // Calculates a remainder
2
3    #include <cs50.h>
4    #include <stdio.h>
5
6    int main(void)
7    {
8        // Prompt user for integer
9        int n = get_int("n: ");
10
11       // Check parity of integer
12       if (n % 2 == 0)
13       {
14           printf("even\n");
15       }
16       else
17       {
18           printf("odd\n");
19       }
20   }
```

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

```c
// Math library

#include <cs50.h>
#include <math.h>
#include <stdio.h>

int main(void)
{
    double base = get_double("Base: ");
    double exponent = get_double("Exponent: ");
    printf("Output: %.0f\n", pow(base, exponent));
}
```

```c
// Return value

#include <stdio.h>

void square(int n);

int main(void)
{
    int input = get_int("Input: ");
    printf("Output: %i\n", square(n));
}

// Square n
int square(int n)
{
    return n * n;
}
```

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