

```
1 // Iterative binary search
2
3 #include <cs50.h>
4 #include <stdio.h>
5 #include <string.h>
6
7 // Names in a phone book
8 string book[] = {
9     "Chen",
10    "Kernighan",
11    "Leitner",
12    "Lewis",
13    "Malan",
14    "Muller",
15    "Seltzer",
16    "Shieber",
17    "Smith"
18 };
19
20 bool search(string name, string names[], int left, int right);
21
22 int main(void)
23 {
24     // Prompt user for name
25     string name = get_string("Name: ");
26
27     // Search for name
28     if (search(name, book, 0, sizeof(book) / sizeof(string) - 1))
29     {
30         printf("Calling %s\n", name);
31     }
32     else
33     {
34         printf("Quitting\n");
35     }
36 }
37
38 // Searches names for name
39 bool search(string name, string names[], int left, int right)
40 {
41     // No more names to search
42     if (left > right)
43     {
44         return false;
45     }
```

```
46
47     // Look at middle
48     int middle = (left + right) / 2;
49     if (strcmp(name, names[middle]) == 0)
50     {
51         return true;
52     }
53
54     // Search left half
55     else if (strcmp(name, names[middle]) < 0)
56     {
57         return search(name, names, left, middle - 1);
58     }
59
60     // Search right half
61     else if (strcmp(name, names[middle]) > 0)
62     {
63         return search(name, names, middle + 1, right);
64     }
65
66     return false;
67 }
```

```
1 // Recursive binary search
2
3 #include <cs50.h>
4 #include <stdio.h>
5 #include <string.h>
6
7 // Names in a phone book
8 string book[] = {
9     "Chen",
10    "Kernighan",
11    "Leitner",
12    "Lewis",
13    "Malan",
14    "Muller",
15    "Seltzer",
16    "Shieber",
17    "Smith"
18 };
19
20 int main(void)
21 {
22     // Prompt user for name
23     string name = get_string("Name: ");
24
25     // Search for name
26     int left = 0, right = sizeof(book) / sizeof(string) - 1;
27     while (left <= right)
28     {
29         // Look at middle
30         int middle = (left + right) / 2;
31         if (strcmp(name, book[middle]) == 0)
32         {
33             printf("Calling %s\n", name);
34             return 0;
35         }
36
37         // Search left half
38         else if (strcmp(name, book[middle]) < 0)
39         {
40             right = middle - 1;
41         }
42
43         // Search right half
44         else if (strcmp(name, book[middle]) > 0)
45         {
```

---

```
46         left = middle + 1;
47     }
48 }
49 printf("Quitting\n");
50 return 1;
51 }
```

```
1  // Extracts a user's initials
2
3  #include <cs50.h>
4  #include <ctype.h>
5  #include <stdio.h>
6  #include <string.h>
7
8  int main(void)
9  {
10     char initials[4];
11     string s = get_string("Name: ");
12     int length = 0;
13     for (int i = 0, n = strlen(s); i < n; i++)
14     {
15         if (isupper(s[i]))
16         {
17             initials[length] = s[i];
18             length++;
19         }
20     }
21     initials[length] = '\0';
22     printf("%s\n", initials);
23 }
```

```
1 // Linear search
2
3 #include <cs50.h>
4 #include <stdio.h>
5 #include <string.h>
6
7 // Names in a phone book
8 string book[] = {
9     "Chen",
10    "Kernighan",
11    "Leitner",
12    "Lewis",
13    "Malan",
14    "Muller",
15    "Seltzer",
16    "Shieber",
17    "Smith"
18 };
19
20 int main(void)
21 {
22     // Prompt user for name
23     string name = get_string("Name: ");
24
25     // Search for name
26     for (int i = 0; i < sizeof(book) / sizeof(string); i++)
27     {
28         if (strcmp(name, book[i]) == 0)
29         {
30             printf("Calling %s\n", name);
31             return 0;
32         }
33     }
34     printf("Quitting\n");
35 }
```

```
1 // Sums a range of numbers iteratively
2
3 #include <cs50.h>
4 #include <stdio.h>
5
6 int sigma(int m);
7
8 int main(void)
9 {
10     int n;
11     do
12     {
13         n = get_int("Positive integer: ");
14     }
15     while (n < 1);
16     int answer = sigma(n);
17     printf("%i\n", answer);
18 }
19
20 // Return sum of 1 through m
21 int sigma(int m)
22 {
23     int sum = 0;
24     for (int i = 1; i <= m; i++)
25     {
26         sum += i;
27     }
28     return sum;
29 }
```

```
1  // Sums a range of numbers recursively
2
3  #include <cs50.h>
4  #include <stdio.h>
5
6  int sigma(int m);
7
8  int main(void)
9  {
10     int n;
11     do
12     {
13         n = get_int("Positive integer: ");
14     }
15     while (n < 1);
16     int answer = sigma(n);
17     printf("%i\n", answer);
18 }
19
20 // Returns sum of 1 through m
21 int sigma(int m)
22 {
23     if (m <= 0)
24     {
25         return 0;
26     }
27     else
28     {
29         return (m + sigma(m - 1));
30     }
31 }
```