

```
1  #include <cs50.h>
2  #include <stdio.h>
3
4  int factorial(int n);
5
6  int main(void)
7  {
8      // Get positive value for N
9      int n;
10     do
11     {
12         n = get_int("n: ");
13     }
14     while (n < 0);
15
16     // Print factorial
17     printf("%i\n", factorial(n));
18 }
19
20 int factorial(int n)
21 {
22     // Base case
23     if (n == 1)
24     {
25         return 1;
26     }
27
28     // Recursive case
29     return n * factorial(n - 1);
30 }
```

```
1  #include <cs50.h>
2  #include <stdio.h>
3
4  typedef struct
5  {
6      string name;
7      int votes;
8  } candidate;
9
10 int main(void)
11 {
12     const int num_candidates = 3;
13     candidate candidates[num_candidates];
14
15     candidates[0].name = "Carter";
16     candidates[0].votes = 10;
17
18     candidates[1].name = "Yuliia";
19     candidates[1].votes = 12;
20
21     candidates[2].name = "Inno";
22     candidates[2].votes = 7;
23
24     // Find highest number of votes
25     int highest_votes = 0;
26     for (int i = 0; i < num_candidates; i++)
27     {
28         if (candidates[i].votes > highest_votes)
29         {
30             highest_votes = candidates[i].votes;
31         }
32     }
33
34     // Print name of candidate with highest number of votes
35     for (int i = 0; i < num_candidates; i++)
36     {
37         if (candidates[i].votes == highest_votes)
38         {
39             printf("%s\n", candidates[i].name);
40         }
41     }
42 }
```