

```
1 // Created by hfwei on 2022/10/13.
2 //
3
4 #include <stdio.h>
5 #include <stdbool.h>
6 #include <math.h>
7 #include <time.h>
8
9 int main() {
10     int max = 0;
11     scanf("%d", &max);
12
13     int count = 0;
14     clock_t start = clock();
15     for (int number = 2; number <= max; number++) {
16         // why 1? why not 0?
17         // int is_prime = 1;
18         // since C99
19         bool is_prime = true;
20
21         for (int i = 2; i * i <= number; i++) {
22             if (number % i == 0) {
23                 // is_prime = 0;
24                 is_prime = false;
25                 break;
26             }
27         }
28
29         if (is_prime) {
30             count++;
31             printf("%d ", number);
32         }
33     }
34     clock_t end = clock();
35
36     printf("\ncount = %d in %f seconds\n",
37           count, (double) (end - start) / CLOCKS_PER_SEC);
38
39     return 0;
40 }
```

```
1 // Created by hfwei on 2022/10/13.
2 //
3
4 #include <stdio.h>
5 int main() {
6     int number = 0;
7     scanf("%d", &number);
8
9     int num_of_digits = 0;
10
11     // "while" version
12 // if (number == 0) {
13 //     num_of_digits = 1;
14 // } else {
15 //     while (number > 0) {
16 //         number /= 10;
17 //         num_of_digits++;
18 //     }
19 // }
20
21 // "do-while" version
22 do {
23     number /= 10;
24     num_of_digits++;
25 } while (number > 0);
26
27 printf("Number of digits is %d.\n", num_of_digits);
28 }
```

```

1 // Created by hfwei on 2022/10/13.
2 //
3
4 #include <stdio.h>
5
6 #define LEN 20
7 int numbers[LEN] = {0};
8
9 int main() {
10     /*
11      * Input the array
12      * Note: fails to run this program in "Run" (Ctrl + D)
13      * See: https://youtrack.jetbrains.com/issue/CPP-5704
14      * Use "Terminal" instead.
15      */
16     int len = -1;
17     while (scanf("%d", &numbers[++len]) != EOF);
18 // while (scanf("%d", &numbers[++len]) == 1);
19
20     for (int i = 0; i < len; i++) {
21         printf("%d ", numbers[i]);
22     }
23     printf("\n");
24
25     printf("-----\n");
26     for (int i = 0; i < len; i++) {
27         // find the minimum of numbers[i .. len - 1]
28         int min = numbers[i];
29         int min_index = i;
30         for (int j = i + 1; j < len; j++) {
31             if (numbers[j] < min) {
32                 min = numbers[j];
33                 min_index = j;
34             }
35         }
36
37         // swap numbers[i] and numbers[min_index]
38         int tmp = numbers[i];
39         numbers[i] = numbers[min_index];
40         numbers[min_index] = tmp;
41
42         // print it out
43         for (int i = 0; i < len; i++) {
44             printf("%d ", numbers[i]);

```

```
45     }  
46     printf("\n");  
47     printf("-----\n");  
48 }  
49 }
```

```
1 // Created by hfwei on 2022/10/13.
2 //
3
4 #include <stdio.h>
5 #include <string.h>
6 #include <stdbool.h>
7
8 #define LEN 21
9 char string[LEN] = "";
10
11 int main() {
12     // example: nolemon,nomelon
13     scanf("%20s", string);
14
15     // int len = 0;
16     // while (string[len] != '\0') {
17     //     len++;
18     // }
19     int len = strlen(string);
20     printf("The length of \"%s\" is %d.\n", string, len);
21
22     bool is_palindrome = true;
23     for (int i = 0, j = len - 1; i < j; i++, j--) {
24         if (string[i] != string[j]) {
25             is_palindrome = false;
26             break;
27         }
28     }
29
30     printf("\"%s\" is %s a palindrome.\n", string,
31           is_palindrome ? "" : "not");
32
33     return 0;
34 }
```

```
1 // Created by hfwei on 2022/10/13.
2 //
3
4 #include <stdio.h>
5
6 #define LEN 10
7 int dictionary[LEN] = {1, 1, 2, 3, 5, 8, 13, 21, 34, 55};
8
9 int main() {
10     int key = 0;
11     scanf("%d", &key);
12
13     int index = -1;
14
15     int low = 0;
16     int high = LEN - 1;
17
18     while (low <= high) {
19         // int mid = low + (high - low) / 2
20         int mid = (low + high) / 2;
21         printf("low = %d\t high = %d\t mid = %d\n", low, high
, mid);
22
23         if (key < dictionary[mid]) {
24             high = mid - 1;
25         } else if (key > dictionary[mid]) {
26             low = mid + 1;
27         } else {
28             index = mid;
29             break;
30             // search for the leftmost one
31             // high = index - 1;
32         }
33     }
34
35     if (index == -1) {
36         printf("Not found!\n");
37     } else {
38         printf("The index of %d is %d.\n", key, index);
39     }
40
41     return 0;
42 }
```

```
1 // file: stars.c
2 // author: hfwei
3 // date: 2022/10/13
4 // description: print stars pyramid
5
6 #include <stdio.h>
7 int main() {
8     int lines;
9     scanf("%d", &lines);
10
11     for (int i = 0; i < lines; ++i) {
12         // print [lines - (i + 1)] spaces
13         for (int j = 0; j < lines - (i + 1); ++j) {
14             printf(" ");
15         }
16
17         // print [2 * i + 1] spaces
18         for (int j = 0; j < 2 * i + 1; ++j) {
19             printf("*");
20         }
21
22         if (i < lines - 1) {
23             printf("\n");
24         }
25     }
26
27     return 0;
28 }
```