```c
// Created by hfwei on 2022/10/13.
//

#include <stdio.h>

//#define LEN INT_MAX
#define LEN 10
int dictionary[LEN] = {1, 1, 2, 3, 5, 8, 13, 21, 34, 55};

int main() {
  int key = 0;
  scanf("%d", &key);

  int index = -1;

  int low = 0;
  int high = LEN - 1;

  while (low <= high) {
    // int mid = low + (high - low) / 2
    int mid = (low + high) / 2;
    printf("low = %d \t high = %d\t mid = %d\n", low, high, mid);
//    printf("mid = %d\n", mid);

    if (key < dictionary[mid]) {
      high = mid - 1;
    } else if (key > dictionary[mid]) {
      low = mid + 1;
    } else {
      index = mid;
      break;
    }
  }

  if (index == -1) {
    printf("Not found!\n");
  } else {
    printf("The index of %d is %d.\n", key, index);
  }

  return 0;
}
```

```
 1 add_executable(2022-stars stars.c)
 2
 3 add_executable(2022-primes primes.c)
 4 target_link_libraries(2022-primes m)
 5
 6 add_executable(2022-binary-search binary-search.c)
 7
 8 add_executable(2022-digits digits.c)
 9
10 add_executable(2022-palindrome palindrome.c)
11
12 add_executable(2022-selection-sort selection-sort.c)
```

```c
// Created by hfwei on 2022/10/13.
//

#include <stdio.h>
int main() {
  int number = 0;
  scanf("%d", &number);

  int num_of_digits = 0;

  // "while" version
//  if (number == 0) {
//    num_of_digits = 1;
//  } else {
//    while (number > 0) {
//      number /= 10;
//      num_of_digits++;
//    }
//  }

  // "do-while" version
  do {
    number /= 10;
    num_of_digits++;
  } while (number > 0);

  printf("Number of digits is %d.\n", num_of_digits);
}
```

```c
// Created by hfwei on 2022/10/13.
//

#include <stdio.h>
#include <string.h>

#define LEN 21
char string[LEN] = "";

int main() {
  scanf("%20s", string);

//  int len = 0;
//  while (string[len] != '\0') {
//    len++;
//  }
  int len = strlen(string);
  printf("The length of \"%s\" is %d.\n", string, len);

  int is_palindrome = 1;
  for (int i = 0, j = len - 1; i < j; i++, j--) {
    if (string[i] != string[j]) {
      is_palindrome = 0;
      break;
    }
  }

  printf("\"%s\" is %s a palindrome.\n", string,
         is_palindrome ? "" : "not");

  return 0;
}
```

```c
// Created by hfwei on 2022/10/13.
//

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

int main() {
  int max = 0;
  scanf("%d", &max);

  for (int number = 2; number <= max; number++) {
    // why 1? why not 0?
    int is_prime = 1;

    for (int i = 2; i < ceil(sqrt(number)); i++) {
      if (number % i == 0) {
        is_prime = 0;
        break;
      }
    }

    if (is_prime) {
      printf("%d ", number);
    }
  }

  return 0;
}
```

```
 1 # 3-for-a-while
 2
 3 ## `stars.c`
 4 - code style:
 5   - empty lines
 6     - `#include`
 7   - [ ] file comments
 8 - `for (int i = 0) + for (int j = 0)`
 9
10 ## `is-prime.c`
11 - `int is_prime = 1;`: why 1? why not 0?
12 - `if (is_prime)` vs. `if (is_prime != 0)` vs. `if (
   is_prime == 1)`
13 - `break`
14 - `i * i < number`
15 - [ ] `stdbool.h`
16
17 # `binary-search.c`
18 - already sorted array
19   - Fib
20 - `int index = -1;`
21   - `printf`
22 - testing
23   - the left-most/right-most one
24 - learn from the standard library
25   - `(low + high) / 2`
26     - try it???
27     - `low + (high - low) / 2`
28
29 ## `digits.c`
30 - testing
31 - `do-while`
32
33 ## `palindrome.c`
34 - `#define`: pre-processing
35 - `scanf("%20s", string);`
36 - `strlen`
37
38 ## `selection-sort.c`
39 - preparation: scanf
40 - with comments
41 - `swap`
42 - [ ] `while (scanf ...)`
43   - K&R C
```

```
44    - How to run this?
45  - more `printf`
46  - `I/O indirection`
```

```c
// Created by hfwei on 2022/10/13.
//

#include <stdio.h>

#define LEN 20
int numbers[LEN] = {0};

int main() {
  int len = -1;
  /*
   * Input the array
   * Note: fails to run this program in "Run" (Ctrl + D)
   * See: https://youtrack.jetbrains.com/issue/CPP-5704
   * Use "Terminal" instead.
   */
  while (scanf("%d", &numbers[++len]) != EOF);

  for (int i = 0; i < len; i++) {
    printf("%d ", numbers[i]);
  }
  printf("\n");

  printf("--------------------\n");
  for (int i = 0; i < len; i++) {
    // find the minimum of numbers[i .. len - 1]
    int min = numbers[i];
    int min_index = i;
    for (int j = i + 1; j < len; j++) {
      if (numbers[j] < min) {
        min = numbers[j];
        min_index = j;
      }
    }

    // swap numbers[i] and numbers[min_index]
    int tmp = numbers[i];
    numbers[i] = numbers[min_index];
    numbers[min_index] = tmp;

    // print it out
    for (int i = 0; i < len; i++) {
      printf("%d ", numbers[i]);
    }
```

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

```c
// file: stars.c
// author: hfwei
// date: 2022/10/13
// description: print stars pyramid

#include <stdio.h>
int main() {
  int lines;
  scanf("%d", &lines);

  for (int i = 0; i < lines; ++i) {
    // print [lines - (i + 1)] spaces
    for (int j = 0; j < lines - (i + 1); ++j) {
      printf(" ");
    }

    // print [2 * i + 1] spaces
    for (int j = 0; j < 2 * i + 1; ++j) {
      printf("*");
    }

    if (i < lines - 1) {
      printf("\n");
    }
  }

  return 0;
}
```