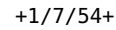


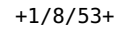
0 .5 1 .5 2 .5

Sachant que les codes ASCII des chiffres 0, 1, 2, ..., 9 sont respectivement 48, 49, 50, ..., 57, qu'affiche le programme ci-dessus ?

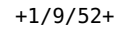


0 .5 1 .5 2 .5

Qu'affiche le programme ci-dessus ?

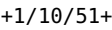


0 .5 1 .5 2 .5



0 .5 1 .5 2 .5

Qu'affiche le programme ci-dessus ?

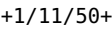


0 .5 1 .5 2 .5

```

1  #include <stdio.h>
2
3  #define N 10
4
5  void my_func(int *p, int *q){
6      int *w = p;
7      if (p != q) {
8          *w = *p;
9          *p = *q;
10         *q = *w;
11     }
12 }
13
14 int main() {
15     int my_array[N] = {1, 9, 3, 7, 4, 5, 6, 8, 2, 0};
16     int my_array_copy[N] = {1, 9, 3, 7, 4, 5, 6, 8, 2, 0};
17
18     int i = 3;
19     do {
20         my_func(my_array + i, my_array + *(my_array + i));
21         i += 1;
22     } while (i < N);
23
24     int res = 0;
25     for (int i = 0; i < N; i++) {
26         res += my_array[i] != my_array_copy[i];
27     }
28     printf("%d %d", my_array[N-2], res);
29     return 0;
30 }

```

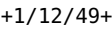


0 .5 1 .5 2 .5

```

1  #include <stdio.h>
2  #include <string.h>
3
4  void my_func_A(char *s1, char *s2) {
5      int i = 0;
6      while (i < strlen(s1)){
7          int res = (s1[i] - '0') && (s2[i] - '0');
8          s2[i] = res != 0 ? '1' : '0';
9          i++;
10     }
11 }
12
13 void my_func_B(char *s1, char *s2, char *s3) {
14     my_func_A(s1, s2);
15     my_func_A(s2, s3);
16 }
17
18 int main() {
19     char p[] = "10101010";
20     char q[] = "11001110";
21     char w[] = "11011011";
22     my_func_B(p, q, w);
23     printf("%s", w);
24     return 0;
25 }

```



0 .5 1 .5 2 .5

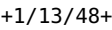
```

1  #include <stdio.h>
2
3  #define N 10
4
5  int val = 1;
6
7  void inc_A(int index, int array[N]){
8      val = index++;
9      array[index] = val++;
10 }
11
12 void inc_B(int index, int array[N]){
13     int val = ++index;
14     array[index] = ++val;
15 }
16
17 int main() {
18     int my_array[N] = {0};
19     inc_A(val, my_array);
20     inc_A(val, my_array);
21     inc_B(val, my_array);
22     inc_B(val, my_array);
23
24     int res = 0;
25     for (int i = 0; i < N; i++) {
26         res += my_array[i];
27     }
28     printf("%d", res);
29     return 0;
30 }

```

Qu'affiche le

programme ci-dessus ?

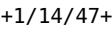


0 .5 1 .5 2 .5

```

1  #include <stdio.h>
2  #include <string.h>
3
4  int main() {
5      char* s[4] = {"The_cat_", "is_black", "and_", "bulldog_is_friendly"};
6      char temp_a[100], temp_b[100];
7      char* c;
8
9      c = strstr(s[3], "is");
10     strcpy(temp_a, *s);
11     strcat(temp_a, c);
12     strcpy(temp_b, *(s + 2));
13     strcat(temp_b, s[1]);
14     if (strcmp(temp_a, temp_b)){
15         printf("%s", temp_b);
16     } else {
17         printf("%s", temp_a);
18     }
19     return 0;
20 }

```



0 .5 1 .5 2 .5