



Санкт-Петербургский
государственный
университет

ПАРАДИГМЫ ПРОГРАММИРОВАНИЯ

1 ИМПЕРАТИВНОЕ

1.2 ПРОЦЕДУРНОЕ

1.3 СТРУКТУРНОЕ

1.4 ООП

2 ДЕКЛАРАТИВНОЕ

2.1 ФУНКЦИОНАЛЬНОЕ





Санкт-Петербургский
государственный
университет

ПАРАДИГМЫ ПРОГРАММИРОВАНИЯ

ИМПЕРАТИВНОЕ

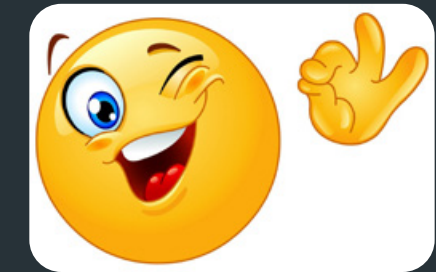
(МАШИННЫЙ ЯЗЫК, ЯЗЫК АССЕМБЛЕРА)

```
1 movf    0x40,W    // Скопировать ячейку с адресом 0x40 (десятичное 64) в регистр W
2 addlw   0x05      // Добавить константу 5 к регистру W
3 movwf   PORTC     // Записать регистр W в выходной порт PORTC микроконтроллера
4 clrw    // Очистить регистр W (у данной инструкции нет операндов)
5 goto    Again     // Перейти на метку Again
```

ПРОЦЕДУРНОЕ

(ПАСКАЛЬ, ФОРТРАН, C)

```
1 #include <iostream>
2 #include <vector>
3 #include <random>
4 #include <time.h>
5 #include <algorithm>
6
7 using namespace std;
8
9 void bubbleSort(vector<int> &arr) {
10     int n = arr.size();
11     for (int i = 0; i < n - 1; i++) {
12         for (int j = 0; j < n - i - 1; j++) {
13             if (arr[j] > arr[j + 1]) {
14                 swap(arr[j], arr[j + 1]);
15             }
16         }
17     }
18 }
19
20 bool checker(vector<int> &arr) {
21     vector<int> arr_copy = arr;
22     sort(arr_copy.begin(), arr_copy.end());
23     bubbleSort(arr);
24     return arr==arr_copy;
25 }
26
27 int main() {
28     int n = 10;
29     vector<int> arr(n);
30     srand(time(NULL));
31     for (int i = 0; i < 100; i++) {
32         for (int j = 0; j < n; j++) {
33             arr[j] = rand();
34         }
35         cout << checker(arr);
36     }
37     return 0;
38 }
```





Санкт-Петербургский
государственный
университет

СТРУКТУРНОЕ ПРОГРАММИРОВАНИЕ

НАУЧНОЕ ПРОГРАММИРОВАНИЕ

- Вычислительные ресурсы
- Сопровождение кода не нужно
- Узко понимаемая эффективность

Главное — это **память и производительность**



1970-е

ПРОМЫШЛЕННОЕ ПРОГРАММИРОВАНИЕ

- Снижение общих затрат на разработку
- Работать в полном соответствии со спецификацией
- Адекватно реагировать на любые действия пользователя
- Должна быть выпущена точно к сроку
- Допускать внесение изменений

Главное — это **надежность** работы программы, возможность **сопровождения** и возможность **планировать производство**