

A1



C++ SELECTION_SORT.cpp

```
1 void selectionSort(std::vector<int> &A) {
2     for (size_t i = 0; i < n; ++i) {
3         int minId = i;
4
5         for (size_t j = i + 1; j < n; ++j) {
6             if (A[j] < A[minId]) {
7                 minId = j;
8             }
9         }
10
11         std::swap(A[minId], A[i]);
12     }
13 }
```

1. Сформулируйте условие $P1$, которое подходит в качестве инварианта внутреннего цикла алгоритма по j . Представьте краткое обоснование (например, с использованием частичной трассировки выполнения цикла).

$P1 : A[\text{minId}] = \min(A[(i + 1) \dots j])$

докажем с метода мат.индукции

во время захода в цикл условие выполняется, потому что это минимум из одного элемента

ШАГ.

Пусть условие выполняется для j , докажем что выполняется для $j + 1$

Если $A[j + 1] < A[\text{minId}]$ тогда $\text{minId} = j + 1$ условие продолжает выполняться

В обратном случае условие тем более продолжает выполняться

ч.т.д.

2. (Сформулируйте условие $P2$, которое подходит в качестве инварианта внешнего цикла алгоритма по i . Представьте краткое обоснование.

$P2$: в конце итерации $A[. . i] = \text{sorted}(A)[. . i]$

докажем методом мат.индукции

БАЗА:

Первый цикл по j найдет наименьший элемент массива, тем самым условие будет выполняться

ШАГ:

пусть условие выполняется для i , докажем для $i + 1$

внутренний цикл найдет наименьший элемент среди $A[i + 1..]$

свопом мы поставим его на место $i + 1$ тем самым инвариант продолжит выполняться

3. Выполните проверку выполнения найденных инвариантов $P1$ и $P2$ до входа в каждый из циклов (INIT), во время итерации циклов (MNT), при выходе из цикла (TRM).

INIT

P1

когда мы входим в цикл, $j = i+1$

тогда в интервале $A[i . (i + 1)]$ будет находится 2 элемента, мы их сравниваем и выбираем меньший. инвариант выполняется

P2

в цикле по j находим индекс минимального элемента, и этот элемент как раз и будет $\text{sorted}(A)[0]$ инвариант выполняется.

MNT

P1

у нас уже есть индекс минимального элемента среди $A[i . j]$

мы сравниваем этот элемент с $A[j + 1]$ и по необходимости меняем minInd , таким образом инвариант выполняется.

P2

$A[. . i]$ уже отсортированы как надо

находим минимальный элемент среди $A[(i + 1)..]$ с помощью цикла по j , и ставим его на место $A[i + 1]$ инвариант продолжает выполняться

TRM

P1

с помощью этого цикла мы находим индекс минимального элемента среди $A[i + 1..]$

тогда при $j = n - 1$ получается что мы нашли минимум на интервале $A[i + 1..n - 1]$, значит инвариант и здесь выполняется

P2

в конце этого цикла наш массив отсортирован по неубыванию

тогда $A = \text{sorted}(A)$ инвариант выполняется!!