Лабораторная работа №3
Поиск суммы, произведения, количества
(2 ак.часа)

Доработать программу из Лабораторной работы №2, подставив вместо заглушки реальный алгоритм поиска решения. При этом:

1) Ввод исходных данных выполнить с клавиатуры, не забывая о приглашениях к вводу.
2) Вывод исходных данных и результатов выполнить на экран вывода консольного приложения, не забывая о пояснениях.
3) Доработайте спецификацию задачи, дополнив ее спецификациями для подзадач(и) решения в сокращенном виде – только описание метода, алгоритм в виде блок-схемы подзадач(и) и программный код, который следует добавить в программу вместо заглушки.
4) **Все** задачи в первом семестре выполнять только с использованием **статических** (не динамических открытых) массивов, и без *goto*, *break*, *exit*. Можно использовать *break*, *return* и *exit*, но не из тела цикла (изучаем циклы, а не break).
5) Проверьте работу программы на ранее составленных тестах, и внесите при необходимости изменения.

Алгоритмы поиска количества, суммы и произведения приведены также в файле Базовые-алгоритмы*.pdf (Base-algorithmesC.pdf)*, ввод и вывод – в файле Кодирование-алгоритмов*.pdf (Algorithm-codingC.pdf)*.

Пример решения задачи описан в *Пример-отчета-для-лабораторной-работы-2.pdf (Lab2-3ExampleC.pdf)*

Перечень задач: см. Лабораторную работу №2

Контрольные вопросы

1. Почему за начальное значение при поиске суммы всех элементов одномерного массива можно принять ноль, а при поиске произведения – единицу, а не ноль?
2. Почему при объединении кода алгоритмов поиска количества (*K*) и суммы (*S*) четных элементов массива *B* из *N* элементов (см. код ниже) нужны операторные скобки *{}* после *if*? Что будет, если их убрать? Почему в этом примере нет необходимости ставить операторные скобки вокруг тела цикла *for* (после второй строки *"{"* и после шестой строки ещё один *"}"*)? Код примера:

Поиск суммы	Поиск количества		Поиск суммы и количества
`S=0;`	`K=0;`	1	`S=K=0;`
`for(i=0; i<N; i++)`	`for (i=0; i<N; i++)`	2	`for (i=0; i<N; i++)`
`  if (B[i] % 2 == 0 )`	`  if (B[i] % 2 == 0 )`	3	`  if (B[i] % 2 == 0) {`
`    S += B[i];`	`    K++;`	4	`      S+= B[i];`
		5	`      K++;`
		6	`    }`

3. Можно ли объединить в одном цикле поиск количества (*K*) четных элементов массива *B* из *N* элементов и суммы (*S*) значений, тех элементов того же массива B, которые кратны 7? Код примера:

Поиск количества	Поиск суммы		Вместе?

`S=0;`
`for(i=0; i<N; i++)`
`  if (B[i] % 2 == 0 )`
`  S += B[i];`
	`K=0;`
	`for (i=0; i<N; i++)`
	`  if (B[i] % 7 == 0 )`
	`  K++;`
		1
		2
		3
		4
		5
		6
			`S=0=K=0;`
			`for (i=0; i<N; i++){`
			`  if (B[i] % 7 == 0) K++;`
			`  if (B[i] % 2 == 0)`
			`  S += B[i];`
			`}`

4. Почему нельзя объединить в одном цикле поиск количества (K) нечетных элементов массива B из N элементов и суммы (S) значений, тех элементов того же массива B, которые кратны (K+2)? В каком порядке их надо включить в код программы? Код примера:

Поиск суммы	Поиск количества	Вместе?

`S=0;`
`for(i=0; i<N; i++)`
`  if (B[i] % (K+2) == 0 )`
`  S += B[i];`
	`K=0;`
	`for (i=0; i<N; i++)`
	`  if (B[i] % 2 != 0 )`
	`  K++;`
		`S=K=0;`
		`for (i=0; i<N; i++){`
		`  if (B[i] % 2 != 0) K++;`
		`  if (B[i] % (K+2) == 0 )`
		`  S += B[i];`
		`}`