

Требования к программам

1. Программа должна получать все параметры в качестве аргументов командной строки.
2. Задачи оцениваются независимо в двух группах: задачи 1–4 и задачи 5–8.
3. Аргументы командной строки для задач 1–3:

- 1) `name` – имя файла,
- 2) `x` – аргумент x (тип `double`).

Например, запуск

```
./a.out a.txt 10
```

означает, что имя файла – это `a.txt`, $x = 10$.

4. Аргументы командной строки для задач 4–8:

- 1) `name` – имя файла.

Например, запуск

```
./a.out a.txt
```

означает, что имя файла – это `a.txt`.

5. Решение задачи должно быть оформлено в виде функции.
6. В функции, реализующей решение задачи, запрещается что-либо выводить на экран.
7. Вывод результата работы в функции `main` для задач 1–2, 8 должен производиться по формату:

```
printf ("%s : Task = %d Result = %d\n",  
        argv[0], task, res);
```

где

- `argv[0]` – первый аргумент командной строки (имя образа программы),
- `task` – номер задачи,
- `res` – возвращаемое значение функции, реализующей решение этой задачи.

8. Вывод результата работы в функции `main` для задачи 3 должен производиться по формату:

```
printf ("%s : Task = %d Result = %d i = %d j = %d\n",  
        argv[0], task, res, i, j);
```

где

- `argv[0]` – первый аргумент командной строки (имя образа программы),
- `task` – номер задачи,

- `res` – возвращаемое значение функции, реализующей решение этой задачи,
- `i, j` – значения переменных `i, j` после работы функции, реализующей решение этой задачи.

9. Вывод результата работы в функции `main` для задачи 4 должен производиться по формату:

```
printf ("%s : Task = %d Result = %d X = %f i = %d j = %d\n",
        argv[0], task, res, x, i, j);
```

где

- `argv[0]` – первый аргумент командной строки (имя образа программы),
- `task` – номер задачи,
- `res` – возвращаемое значение функции, реализующей решение этой задачи,
- `x, i, j` – значения переменных `x, i, j` после работы функции, реализующей решение этой задачи.

10. Вывод результата работы в функции `main` для задач 5–7 должен производиться по формату:

```
printf ("%s : Task = %d Result = %d X = %f\n",
        argv[0], task, res, x);
```

где

- `argv[0]` – первый аргумент командной строки (имя образа программы),
- `task` – номер задачи,
- `res` – возвращаемое значение функции, реализующей решение этой задачи,
- `x` – значения переменной `x` после работы функции, реализующей решение этой задачи.

Задачи

1. Написать функцию, получающую в качестве аргументов вещественное число `x` и имя файла, содержащего последовательность x_1, x_2, \dots вещественных чисел неизвестной длины, и возвращающую целое число, равное номеру первого элемента этой последовательности, совпадающего с `x`, если такой существует, и 0, если такого элемента нет. Функция должна возвращать `-1, -2` и т.д., если она не смогла открыть файл, прочитать элемент и т.д..
2. Написать функцию, получающую в качестве аргументов вещественное число `x` и имя файла, содержащего последовательность x_1, x_2, \dots вещественных чисел неизвестной длины, и возвращающую целое число, равное номеру последнего элемента этой последовательности, совпадающего с `x`, если такой существует, и 0, если такого элемента нет. Функция должна возвращать `-1, -2` и т.д., если она не смогла открыть файл, прочитать элемент и т.д..
3. Написать функцию, получающую в качестве аргументов вещественное число `x`, адреса целых переменных `i, j` и имя файла, содержащего последовательность x_1, x_2, \dots вещественных чисел неизвестной длины. Функция возвращает целое число, равное 1, если `x` присутствует в этой последовательности, и 0 в противном случае. Если `x` есть в последовательности, то целая переменная `i` получает значение, равное номеру первого элемента последовательности, совпадающего с `x`, переменная `j` получает значение, равное номеру последнего элемента последовательности, совпадающего с `x`. Если числа `x` в последовательности нет, то переменные `i` и `j` не изменяются. Функция должна возвращать `-1, -2` и т.д., если она не смогла открыть файл, прочитать элемент и т.д., при этом переменные `i, j` не изменяются.

4. Написать функцию, получающую в качестве аргументов адреса вещественной переменной x и целых переменных i, j , и имя файла, содержащего последовательность x_1, x_2, \dots вещественных чисел неизвестной длины. В результате работы функции вещественная переменная x получает значение, равное максимальному элементу этой последовательности. Целая переменная i получает значение, равное номеру первого максимального элемента в последовательности, переменная j получает значение, равное номеру последнего максимального элемента в последовательности. Функция должна возвращать 0 в случае успеха и $-1, -2$ и т.д., если она не смогла открыть файл, прочитать элемент и т.д., при этом переменные x, i, j не изменяются.
5. Написать функцию, получающую в качестве аргументов адрес вещественной переменной x и имя файла, содержащего последовательность x_1, x_2, \dots вещественных чисел неизвестной длины, и возвращающую целое число, равное максимальной длине участка постоянства в последовательности. Участком постоянства называются идущие подряд одинаковые элементы в количестве более одного. Длиной участка называется количество элементов в нем. Вещественная переменная x получает значение, равное среднему арифметическому элементов, находящихся во всех участках постоянства этой последовательности с этой максимальной длиной. Функция должна возвращать $-1, -2$ и т.д., если она не смогла открыть файл, прочитать элемент и т.д., при этом переменная x не изменяется.
6. Написать функцию, получающую в качестве аргументов адрес вещественной переменной x и имя файла, содержащего последовательность x_1, x_2, \dots вещественных чисел неизвестной длины, и возвращающую целое число, равное количеству участков постоянства в последовательности. Участком постоянства называются идущие подряд одинаковые элементы в количестве более одного. Вещественная переменная x получает значение, равное среднему арифметическому элементов, находящихся во всех участках постоянства этой последовательности. Функция должна возвращать $-1, -2$ и т.д., если она не смогла открыть файл, прочитать элемент и т.д., при этом переменная x не изменяется.
7. Написать функцию, получающую в качестве аргументов имя файла, содержащего последовательность вещественных чисел неизвестной длины, и адрес вещественного числа x , и возвращающую в x вещественное число, равное среднему квадратическому отклонению чисел из этого файла от их среднего арифметического. Возвращаемое значение функции равно длине последовательности в случае успешного завершения и равно $-1, -2$ и т.д., если она не смогла открыть файл, прочитать элемент и т.д., при этом переменная x не меняется.
8. Написать функцию, получающую в качестве аргумента имя файла, содержащего последовательность вещественных чисел неизвестной длины, и возвращающую целое число, равное 1, если эта последовательность является арифметической прогрессией, 2, если она является геометрической прогрессией, 3, если постоянна, 4, если в ней недостаточно элементов для принятия решения, и 0 в противном случае. Функция должна возвращать $-1, -2$ и т.д., если она не смогла открыть файл, прочитать элемент и т.д..