

## Описание абстрактной машины

Машина содержит два регистра —  $a$  и  $d$  — и обрабатывает входной поток, который представляет собой список целых неотрицательных чисел.

Регистр  $d$  может хранить целые неотрицательные числа, регистр  $a$  — целые неотрицательные плюс еще одно специальное значение  $\perp$ .

Машина управляется программой, которая является последовательностью инструкций, каждая из которых может быть снабжена не более чем одной меткой. В результате выполнения программы машина либо выдает одно целое неотрицательное число, либо аварийно останавливается.

Инструкции программы и их семантика таковы:

1.  $e$  — если  $a = \perp$  и входной поток пуст, то завершает выполнение программы с результатом  $d$ , иначе аварийно останавливает машину;
2.  $r$  — если  $a = \perp$  и входной поток непуст, то читает из входного потока очередное число (входной поток при этом становится на одно значение короче), записывает его в  $a$  и переводит машину к исполнению следующей инструкции, в противном случае машина аварийно останавливается;
3.  $j\ n\ l$  (где  $n$  — неотрицательное целое число,  $l$  — метка в программе) — если  $a = 0$ , то устанавливает  $a = \perp$ , прибавляет  $n$  к  $d$  и переводит машину к исполнению инструкции, помеченной меткой  $l$ , иначе, если  $a > 0$ , то уменьшает  $a$  на единицу и переводит машину к исполнению следующей инструкции, иначе аварийно останавливает машину.

Исполнение программы начинается с первой инструкции в состоянии  $a = \perp, d = 0$ .

Программы представляются текстовым файлом, в котором инструкции идут одна за другой, каждая на своей строке. Аргументы инструкций отделяются друг от друга пробелами или символами табуляции, метка от помеченной ею инструкции дополнительно отделяется двоеточием.

Например, следующая программа

```

                r
                j 5 11
11:             r
                j 21 12
12:             e
```

на входном потоке 0 0 возвращает 26, а на входном потоке 0 аварийно останавливается.

## Первая задача

Написать интерпретатор `int` абстрактной машины. Интерпретатор получает в качестве аргументов имя файла с программой и входной поток и

печатает на стандартный вывод число-результат либо символ “.”, если машина аварийно остановилась. Например, если программа из предыдущего примера находится в файле “test.p”, то

```
int test.p 0 0
```

печатает 26, а

```
int test.p 0
```

печатает “.”

Результат оформить в виде пул-реквеста в репозиторий, должен проходить первый набор тестов. Решение должно быть приятно выглядящей программой, использующей алгебраические типы данных и всё такое. Лапша на строках, парах и т.д., гадкий длинный код на “костылях” приниматься не будет. Максимальная оценка — 4.

## Вторая задача

Написать утилиту `analyze`, которая принимает на вход имя файла с программой и определяет, на каком входе данная программа дает наименьший результат. Предполагается, что программа нетривиальна (то есть не останавливается аварийно на всех входах). Результатом программы является текстовое представление значения типа `(Int, [Int])`. Таким образом,

```
analyze test.p
```

должна напечатать `(26, [0,0])`.

Требования к форме отчетности и тексту реализации аналогичные. Решение данной задачи дает возможность получить оценку 5.

## Примечание

*Списывание будет караться по законам экзаменационного времени.*