

## 3.2 Параллельная обработка

---

### Параллельная обработка

*По данным  $n$  процессорам и  $m$  задач определите, для каждой из задач, каким процессором она будет обработана.*

**Вход.** Число процессоров  $n$  и последовательность чисел  $t_0, \dots, t_{m-1}$ , где  $t_i$  — время, необходимое на обработку  $i$ -й задачи.

**Выход.** Для каждой задачи определите, какой процессор и в какое время начнёт её обрабатывать, предполагая, что каждая задача поступает на обработку первому освободившемуся процессору.

---

В данной задаче ваша цель — реализовать симуляцию параллельной обработки списка задач. Такие обработчики (диспетчеры) есть во всех операционных системах.

У вас имеется  $n$  процессоров и последовательность из  $m$  задач. Для каждой задачи дано время, необходимое на её обработку. Очередная работа поступает к первому доступному процессору (то есть если доступных процессоров несколько, то доступный процессор с минимальным номером получает эту работу).

**Формат входа.** Первая строка входа содержит числа  $n$  и  $m$ . Вторая строка содержит числа  $t_0, \dots, t_{m-1}$ , где  $t_i$  — время, необходимое на обработку  $i$ -й задачи. Считаем, что и процессоры, и задачи нумеруются с нуля.

**Формат выхода.** Выход должен содержать ровно  $m$  строк:  $i$ -я (считая с нуля) строка должна содержать номер процессора, который получит  $i$ -ю задачу на обработку, и время, когда это произойдёт.

**Ограничения.**  $1 \leq n \leq 10^5$ ;  $1 \leq m \leq 10^5$ ;  $0 \leq t_i \leq 10^9$ .

**Пример.**

Вход:

```
2 5
1 2 3 4 5
```

Выход:

```
0 0
1 0
0 1
1 2
0 4
```

**Пример.**

Вход:

```
4 20
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1
```

Выход:

```
0 0
1 0
2 0
3 0
0 1
1 1
2 1
3 1
0 2
1 2
2 2
3 2
0 3
1 3
2 3
3 3
0 4
1 4
2 4
3 4
```