Documentación XXXPROGRAMACION

Estructura programa del tipo “n ejecuciones”:

```java
public static void main(String[] args) {
    int n = fastIO.nextInt();
    for (int i = 0; i < n; i++) {
        problem();
        //fastIO.flush();
    }
    fastIO.close();
}
```

Estructura programa del tipo “hasta que la entrada sea x”:

```java
public static void main(String[] args) {
    while (problem()) {
        //fastIO.flush();
    }
    fastIO.close();
}
```

Estructura programa del tipo “hasta que termine la entrada”:

```java
public static void main(String[] args) {
    while (fastIO.hasNext()/*true*/) {
        problem();
        //fastIO.flush();
    }
    fastIO.close();
}
```

Entrada y salida en Java:

```java
static class FastIO {
    BufferedReader br;
    StringTokenizer st;
    BufferedWriter bw;

    public FastIO() {
        br = new BufferedReader(new InputStreamReader(System.in));
        bw = new BufferedWriter(new OutputStreamWriter(System.out));
    }
```

```
    String next() {
        while (st == null || !st.hasMoreElements()) {
            try { st = new StringTokenizer(br.readLine()); }
            catch (IOException e) { e.printStackTrace(); }
        }
        return st.nextToken();
    }

    int nextInt() {
        return Integer.parseInt(next());
    }

    long nextLong() {
        return Long.parseLong(next());
    }

    double nextDouble() {
        return Double.parseDouble(next());
    }

    String nextLine() throws IOException {
        String line = "";
        try { line = br.readLine(); }
        catch (IOException e) { e.printStackTrace(); }
        return line;
    }

    void print(String s) {
        try { bw.write(s); }
        catch (IOException e) { e.printStackTrace(); }
    }

    void println(String s) {
        print(s);
        try { bw.newLine(); }
        catch (IOException e) { e.printStackTrace(); }
    }

    void close() {
        try {
            br.close();
            bw.close();
        } catch (IOException e) {
            e.printStackTrace();
        }
    }
}
```