目的：输入一个XML文档，输出一个事件序列（目前的事件序列是什么还待定）。

方法：分阶段解决问题

1. 先解决单线程版本。
2. 首先将文件读入，获取每一个’<’之后，确定是什么类型，并且要记录开始以及结束位置。若是CDATA、COMMENT、PI之一，则内容中要容许存在’<’；若是其他的TAG，不容许存在‘<’。每一个数据块大小不超过DATA\_SET\_MAX（初定为10k），若截断数据块，则将数据块的尾指针回溯至上一个’<’处。将划分的结果存入数据块的头数据结构中。若截断的是CDATA、COMMENT、PI之一，在上面的处理过程中就应该有退出的逻辑。

若回溯时发现上一个’<’不存在，报错“DATA\_SET\_MAX too small”，希望得到更大的DATA\_SET\_MAX。

1. 对每一个数据块做解析，解析的方法是根据TAG的类型分别解析，具体实现可以参照标准文档。解析过程中如果发现了没有STAG的ETAG，就压入queue中；若发现没有ETAG的STAG，就压入stack中。

解析的元素有六种：

CDATA、COMMENT、PI三种直接处理内容即可。使用的函数模板相同。

STAG和EMPTYTAG一并处理，其中EMPTYTAG不需要验证有没有对一个的ETAG。

ETAG判断当前数据块的上一个STAG是否匹配，若为空，就加入queue。

1. 解析完之后，若queue和stack非空，同时从栈头以及队头弹出一个元素，进行一一匹配，若匹配失败，暂时先报错。随后将所有的数据块顺序输出即可。
2. 流水线多线程

在2和3之间建立一个二级流水线，两个线程同时处理，每个线程只处理本阶段的工作

1. 1的SIMD优化

在判断‘<‘号的时候可以一次处理16个字符哦。

1. 事件多线程

2的多线程处理，做一个线程池，将所有的未处理的事件并行处理，最后串行输出。

1. 优化

后续部分的优化。