

Algorithm 1: A part of Transformation Algorithm

Input: *BPMN collaboration model in XML*

Output: *CSPCode*

```
1 BPMNodes, MsgFlow, SeqFlow = ParseBPMNModel(); // parsing the BPMN
   collaboration model, getting the elements related to interactions
   (stored in BPMNodes), MessageFlow (stored in MsgFlow) and Sequence Flow
   (Stored in SeqFlow)
2 for msgflow in MsgFlow do // generating the code that defines messages and
   channels
   | // generating the code that defines a message for msgflow
3   MsgCode += CodeForMsgDef(msgflow);
   | // generating the code that defines a message channel for msgflow
4   ChCode += CodeForChDef( msgflow);
5 CSPCode += MsgCode;
6 CSPCode += ChCode;
   // get the id for each participant process
7 participantProcesses = getParticipantProcessID(BPMNodes);
8 for p in participantProcesses do
   | // generating the csp# code for each participant
9   element = getStartEvent(p, BPMNodes); // get the start event for p
10  while element != null do
11    | if element.type is startEvent then
12    | | element = SeqFlow[element];
13    | | continue;
14    | if element.type is one of the TaskElements then
15    | | // generate code for the task according to Table 2
16    | | ProcCode += CodeForTask(element);
17    | | element = SeqFlow[element];
18    | | continue;
19    | if element.type is one of the Gateways then
20    | | // generate code for the tasks in the gateway according to
21    | | Table 2
22    | | element, ProcCode = CodeForTasksInGateway(element, ProcCode);
23    | | continue;
24  CSPCode += ProcCode;
25 return CSPCode;
```
