

class 消費者

instance variables

$b : Buffer;$

$producer : 生産者;$

operations

public

消費者 : $Buffer \xrightarrow{o}$ 消費者

消費者 ($aBuffer$) \triangleq

($b := aBuffer;$

return self

);

項目消費 : $char^* \xrightarrow{o} ()$

項目消費 (c) \triangleq

let - = new IO ().echo (" ThreadID = " \curvearrowright Integer.asString (threadid) \curvearrowright ", ConsumeData = " \curvearrowright c \curvearrowright ", \n") in
skip;

public

doit : $() \xrightarrow{o} ()$

doit () \triangleq

while true

do (let $c = b.Get ()$ in

if $c = 'Z'$

then return

else 項目消費([c])

);

public

wait : $() \xrightarrow{o} ()$

wait () \triangleq

skip

sync

per wait \Rightarrow #fin (doit) > #act (wait)

thread

```

        doit()
end 消費者
Test Suite :      vdm.tc
Class :          消費者

```

Name	#Calls	Coverage
消費者 ‘消費者	1	✓
消費者 ‘doit	1	✓
消費者 ‘wait	1	✓
消費者 ‘項目消費	6	✓
Total Coverage		100%