

Test machine

Klaus Grue

GRD-2009-11-28.UTC:11:19:07.179148

Contents

1 Definition of test machine

1

1 Definition of test machine

The test1 machine with handler h_1 defines a new request $r_1(\dots)$ using an extend request and then exercises the new request.

```
[m1 ≐ ‘
#define TRUE 1
#define FALSE 0
#include <stdio.h >
int lgw2dll=TRUE;
unsigned char buffer[]="World\n";
int bufferpnt;
int trigger(int c){
if(lgw2dll){
if(c < 0){lgw2dll=FALSE; bufferpnt=0; }else{printf("%c", c); return
0; }}
if(buffer[bufferpnt]==0) return -1;
return buffer[bufferpnt++]; }
’]

[r1(x) ≐ ⟨[r1(x)]h, x⟩]

[h1 ≐ map ( λx.M1(x) )]

[M1(x) ≐ ⟨extend request ( r1(T) , m1 ), r1(‘Hello’), exec request (
T , handler2 )⟩]

[handler2 ≐ map ( λx.machine2 ( x ) )]

[machine2 ( x ) ≐
if x ∈ A then ⟨write request ( Notfound )⟩ else
if not xh  $\stackrel{t}{=} r_1(T)$  then machine2 ( xt ) else
⟨write request ( xhth ), quit request ( 0 )⟩]
```

$[\text{'test1'} \stackrel{\text{exec}}{=} \langle h_1 \rangle]$

For more, see [1].

References

- [1] Klaus Grue. Test machine - appendix. Technical report, Logiweb, 2008. [.../.../logiweb/01CA9680910201CFF3348602C21BF1B0E67EFD09308C92EAB6BBD8BB0806/page/appendix.pdf](http://.../logiweb/01CA9680910201CFF3348602C21BF1B0E67EFD09308C92EAB6BBD8BB0806/page/appendix.pdf).