In Section 2.2 there is a self-describing expression
```
""""0;0;(0;..29);28;28;(1;..28) """"0;0;(0;..29);28;28;(1;..28)
```
which evaluates to its own representation.

(a) Write an expression that evaluates to twice its own representation. In other words, it evaluates to its own representation followed by its own representation again.
§
```
""""2*(0;0;(0;..33);32;32;(1;..32)) """"2*(0;0;(0;..33);32;32;(1;..32))
```

(b) Make it into a self-printing program. Let's say that !e prints the value of expression e.
§
```
""""!""""0;1;0;1;(1;..34);33;33;(2;..33) """"!""""0;1;0;1;(1;..34);33;33;(2;..33)
```

Here is a program that prints itself twice with a period between (for sequential composition).
```
""""!""""0;1;0;1;(1;..76);75;76;75;(1;..76);76;0;1;0;1;(1;..76);75;76;75;(2;..75) """"
```
```
0;1;0;1;(1;..76);75;76;75;(2;..75);76;0;1;0;1;(1;..76);75;76;75;(2;..75)
```
```
```
```

When this program is executed, it prints a program that's twice as long. And when that program is executed, it prints a program that's four times as long as the original. And so on, with exponentially increasing length.