What Do We Return?

```
int32_t print_reverse
     (const char* s)
{
     int32_t rest;
     if ('\0' == *s) {
          return 0;
     }
     rest = print_reverse (s + 1);
     printf ("%c", *s);
     return (rest + 1);
}
```

What should be returned?

Reference Version of **print_reverse**

```
int32_t print_reverse(const char* s)
{
     int32_t rest;
     if ('\0' == *s) {
          return 0;
     }
     rest = print_reverse (s + 1);
     printf ("%c", *s);
     return (rest + 1);
}
```

Code is available on the web page.

Let’s See How a Recursive Function Works in Detail

Let’s execute a call:

print_reverse ("Now")

With … stack frames!

This visualization will probably be the last time that we see stack frames in class.

Feel free to get nostalgic.

Stack Frame for **print_reverse** (Call Depth 1)

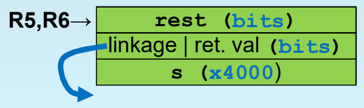
call depth 1 🟩

`p_r ("Now")`

Abbreviated as **`p_r`** here.

Say that the **`N`** is at x4000.

What are the values when the function starts?


R5,R6→
rest (bits)
linkage | ret. val (bits)
s (x4000)