

McCarthy 91 function

The **McCarthy 91 function** is a recursive function, defined by computer scientist John McCarthy as a test case for formal verification within computer science.

The McCarthy 91 function is defined as

$$M(n) = \begin{cases} n - 10, & \text{if } n > 100 \\ M(M(n + 11)), & \text{if } n \leq 100 \end{cases}$$

The results of evaluating the function are given by $M(n) = 91$ for all integer arguments $n \leq 101$, and $M(n) = n - 10$ for $n > 101$.

History

The 91 function was introduced in papers published by Zohar Manna, Amir Pnueli and John McCarthy in 1970. These papers represented early developments towards the application of formal methods to program verification. The 91 function was chosen for having a complex recursion pattern (contrasted with simple patterns, such as defining $f(n)$ by means of $f(n - 1)$). The example was popularized by Manna's book, *Mathematical Theory of Computation* (1974). As the field of Formal Methods advanced, this example appeared repetitively in the research literature. In particular, it is viewed as a "challenge problem" for automated program verification.

Often, it is easier to reason about non-recursive computation. As one of the examples used to demonstrate such reasoning, Manna's book includes a non-recursive algorithm that simulates the original (recursive) 91 function. Many of the papers that report an "automated verification" (or termination proof) of the 91 function only handle the non-recursive version.

A formal derivation of the non-recursive version from the recursive one was given in a 1980 article by Mitchell Wand, based on the use of continuations.

Examples

Example A:

```
M(99) = M(M(110)) since 99 ≤ 100
      = M(100)    since 110 > 100
      = M(M(111)) since 100 ≤ 100
      = M(101)    since 111 > 100
      = 91        since 101 > 100
```

Example B:

```
M(87) = M(M(98))
      = M(M(M(109)))
      = M(M(99))
      = M(M(M(110)))
      = M(M(100))
      = M(M(M(111)))
      = M(M(101))
      = M(91)
      = M(M(102))
      = M(92)
      = M(M(103))
```

```

    = M(93)
... Pattern continues
    = M(99)
(same as example A)
    = 91

```

Code

Here is how John McCarthy may have written this function in Lisp, the language he invented:

```

(defun mc91 (n)
  (cond ((<= n 100) (mc91 (mc91 (+ n 11))))
        (t (- n 10))))

```

Here is an implementation of the non-recursive algorithm in C:

```

int mccarthy(int n)
{
    for (int c = 1; c != 0; ) {
        if (n > 100) {
            n = n - 10;
            c--;
        } else {
            n = n + 11;
            c++;
        }
    }
    return n;
}

```

Proof

Here is a proof that the function behaves as expected.

For $90 \leq n < 101$,

```

M(n) = M(M(n + 11))
     = M(n + 11 - 10), where n + 11 >= 101 since n >= 90
     = M(n + 1)

```

So $M(n) = 91$ for $90 \leq n < 101$.

We can use this as a base case for induction on blocks of 11 numbers, like so:

Assume that $M(n) = 91$ for $a \leq n < a + 11$.

Then, for any n such that $a - 11 \leq n < a$,

```

M(n) = M(M(n + 11))
     = M(91), by hypothesis, since a ≤ n + 11 < a + 11
     = 91, by the base case.

```

Now by induction $M(n) = 91$ for any n in such a block. There are no holes between the blocks, so $M(n) = 91$ for $n < 101$. We can also add $n = 101$ as a special case.

Knuth's generalization

Donald Knuth generalized the 91 function to include additional parameters. John Cowles developed a formal proof that Knuth's generalized function was total, using the ACL2 theorem prover.

References

- Zohar Manna and Amir Pnueli (July 1970). "Formalization of Properties of Functional Programs". *Journal of the ACM* **17** (3): 555–569. doi:10.1145/321592.321606.
- Zohar Manna and John McCarthy (1970). "Properties of programs and partial function logic". *Machine Intelligence* **5**.
- Zohar Manna. *Mathematical Theory of Computation*. McGraw-Hill Book Company, New-York, 1974. Reprinted in 2003 by Dover Publications.
- Mitchell Wand (January 1980). "Continuation-Based Program Transformation Strategies". *Journal of the ACM* **27** (1): 164–180. doi:10.1145/322169.322183.
- Donald E. Knuth (1991). "Textbook Examples of Recursion". arXiv:cs/9301113.
- John Cowles (2000). "Knuth's generalization of McCarthy's 91 function" ^[1]. *Computer-Aided reasoning: ACL2 case studies*. Kluwer Academic Publishers. pp. 283–299.

References

- [1] <http://www.cs.utexas.edu/users/moore/ac12/workshop-1999/Cowles-abstract.html>

Article Sources and Contributors

McCarthy 91 function *Source:* <http://en.wikipedia.org/w/index.php?oldid=419848037> *Contributors:* AHM, AmirOnWiki, Anog, Arthena, Brentsmith101, CBM, David Eppstein, Dysprosia, Frencheigh, Headbomb, Jamelan, Joseph Solis in Australia, Mrusof, Mshonle, Nick8325, Oleg Alexandrov, Pchugh, Piet Delpport, Poor Yorick, Quarkflavour, Rheun, SarekOfVulcan, 21 anonymous edits

License

Creative Commons Attribution-Share Alike 3.0 Unported
<http://creativecommons.org/licenses/by-sa/3.0/>
