

What code is beautiful?

Beauty is in the eye of the beholder
The various cultures of programming
have different ideas about code
and write it for different purposes.
Their aesthetic judgment differs too.

5. The Semantics of Microalgol

The semantics of Microalgol is given by a recursively defined function $\text{micro}(\pi, \xi)$ that gives the state in which a Microalgol program π will terminate if it is entered in state ξ .

We now can write

```
micro( $\pi, \xi$ ) = ( $\lambda n$ .if end( $\pi, n$ ) then  $\xi$ 
else ( $\lambda s$ .if assignment( $s$ ) then
micro( $\pi, a(s, n + 1, a(\text{left}(s), \text{value}(\text{right}(s), \xi), \xi))$ )
else if goto( $s$ ) then
micro( $\pi, a(s, n, \text{value}(\text{proposition}(s), \xi)$ ) then
(statement( $n, \pi$ )) (c( $s, \xi$ ))
```

Hacker culture

Clever code is the hallmark of a good hacker.
The inverse square root function from the Quake III game uses clever math and relies on intricate aspects of the bit representation of floating point numbers.

This completes the description of abstract Microalgol.

```
float Q_rsqrt( float number )
{
    long i;
    float x2, y;
    const float threehalfs = 1.5F;

    x2 = number * 0.5F;
    y = number;
    i = * ( long * ) &y;           // evil floating point bit level hacking
    i = 0x5f3759df - ( i >> 1 );   // what the fuck?
    y = * ( float * ) &i;
    y = y * ( threehalfs - ( x2 * y * y ) ); // 1st iteration
    //y = y * ( threehalfs - ( x2 * y * y ) ); // 2nd iteration, this can be removed
    return y;
}
```

5. The Semantics of Microalgol

The semantics of Microalgol is given by a recursively defined function $\text{micro}(\pi, \xi)$ that gives the state in which a Microalgol program π will terminate if it is entered in state ξ .

We now can write

```
micro( $\pi, \xi$ ) = ( $\lambda n$ .if end( $\pi, n$ ) then  $\xi$ 
else ( $\lambda s$ .if assignment( $s$ ) then
micro( $\pi, a(s, n + 1, a(\text{left}(s), \text{value}(\text{right}(s), \xi), \xi))$ )
else if goto( $s$ ) then
micro( $\pi, a(s, n, \text{value}(\text{proposition}(s), \xi)$ ) then
numb(destination( $s$ ),  $\pi$ ) else  $n + 1, \xi$ )))
(statement( $n, \pi$ )) (c( $s, \xi$ ))
```

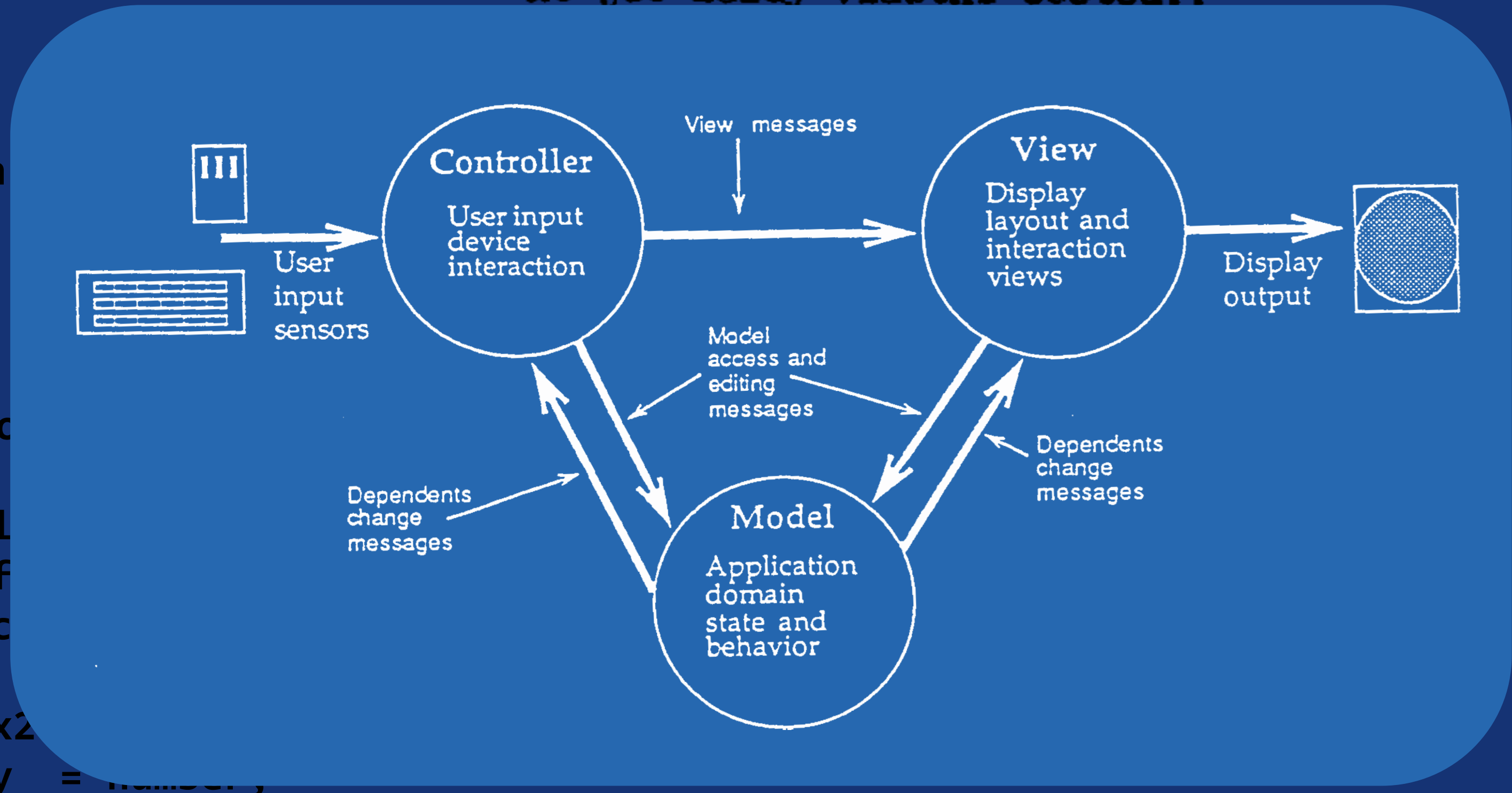
This completes the description of abstract Microalgol.

Mathematical culture

A programming language is beautiful if it achieves great expressivity using a small number of orthogonal features.
The elegant structure of Algol 60 allowed John McCarthy to write a concise formal definition of the language.

Engineering culture

Beautiful code is simple and well-structured. Readability is valued more than cleverness. The Model-View-Controller pattern introduced in Smalltalk-80 separated different functions of the user interface and enabled reusability.



Double coding

Code has a dual role.
It is read by a human and executed by a computer.

Code poets and hackers exploit this double meaning, while engineers try to keep both meanings aligned.

A similar double coding is at play in Václav Havel's poem ("Let everyone go their own way").

Humanistic culture

Code is written not just to run, but also to speak to a human. The poem water.c evokes streams of water when read as a poem and at the same time generates a rainfall effect when run as code.

```
#define submerge const char* _=0%239?" ":"\t;\t";0*=2654435761;int
#define _cOb8(...) int s,on,__VA_ARGS__;int main(int 0, char**Q)

_cOb8(o_, _o08ocQ0cOb, _ocQbo8oo, _o08ocOb_
){ ; { ; ; ; } ; { ; ; ; } ; { ; ; ; } ; { ; ; ; }
; ; break; the; ; ; long grass ;unsigned squall ; ;
} } ; while (1){soft; submerge us; ; in: sleep (0) ;
; ; printf (_); quietly :on ; ; the; soil; ; }
{ ; ; ; ; ; ; ; ; ; ; ; ; ; ; ; ; ; ; ; ; ; ;
{ ; shake: time (1) ;register *_ , the =clock(s ) ;
; ; volatile *_ , winds ; ; double wills ;{
; char the ,* fire ; ; short companion,* ;}
; { union {}* , together ; ; ; void *warms ;}
; ; ; ; ; ; ; ; ; ; ; ; ; ; ; ; ; ; ; ; ; ;
; ; if (1) wet ; ; raise (1); struct{;ure ; ; ;
; ; free (0);for(;;){ newborn ; ; daughter ; ; ;
; ; extern al, ** world ,*re;const ructed ; ; ;
; ; ; continue;on;; floods ; ; ; of: water ; ; ; }
; ; ; ; ; ; ; ; ; ; ; ; ; ; ; ; ; ; ; ; ; ; }
```


Page 3

Water <https://code-poetry.com/water>

MVC <https://iihm.imag.fr/blanch/ens/2006-2007/RICM3/IHM/documents/krasner-MVC.pdf>

Quake 3 <https://github.com/id-Software/Quake-III-Arena>