
	<i>Program Size</i> (Kbytes)	<i>Analysis Time</i> (sec)	<i>Analysis Space</i> (Mbytes)
Boyer	21.4	8.3	13.8
Browse	8.3	1.5	2.2
Takr	17.5	3.8	6.5
Traverse	5.2	1.1	1.6
Dtype	69.0	17.7	22.7
Interp	8.6	1.5	3.3

Figure 7.4 Time and Space Requirements

```
(define (fft areal aimag)
  (let ([ar areal]
        [ai aimag])
    :
    many uses of ar, ai via vector-ref, vector-set!
```

our type checker is able to assign type (*vec num*) to both **ar** and **ai**. For this modified program, Soft Scheme inserts no run-time checks at all.

7.5.2 Div

Div contains a procedure that constructs lists of length n :⁶

```
(define create-n
  (lambda (n)
    (let loop ([n n][a '()])
      (if (= n 0)
          a
          (loop (- n 1) (cons '() a))))))
```

Soft Scheme infers type ($num \rightarrow list\ nil$) for this procedure. But the program only calls **create-n** with an even number, and other procedures assume that the lists con-

⁶We have taken the liberty of rewriting this procedure to avoid Scheme's **do**-construct.