

1 Assignment, Optimizing Compilers SS 2003

```
minimum (array, n)
{
    min = 0;
    next = 0;
    null_size = n <= 0;
    if (null_size) goto EXIT;
    min = *(array);
    upper_bound = n - 1;
LOOP:
    end_reached = next >= upper_bound;
    if (end_reached) goto EXIT;
    next = next + 1;
    addr = SOL * next;
    addr = addr + array;
LOOP1:
    tmp = *(addr);
    tag = min < tmp;
    if (tag) goto LOOP;
    min = tmp;
    goto LOOP;
FIND:
    if (tag) goto LOOP
    goto EXIT;
EXIT:
    return min;
}
```

Figure 1: Example in OIL

1.1 Find all basic blocks

1.2 Construct CFG

1.3 Compute predecessors and successors

1.4 Find unreachable basic blocks

NB: List all iteration steps of the work-list algorithm.