

## 编译原理和技术 作业 7

**习题 9.3.** 对图 9.32 的流图, 计算:

- (a) 为到达-定值分析, 计算每个块的  $gen$ ,  $kill$ , IN 和 OUT 集合.
- (b) 为可用表达式分析, 计算每个块的  $e\_gen$ ,  $e\_kill$ , IN 和 OUT 集合.
- (c) 为活跃变量分析, 计算每个块的  $def$ ,  $use$ , IN 和 OUT 集合.

**解:** (a) 先求  $gen$  和  $kill$  集合, 如下

基本块	$gen$	$kill$
$B_1$	$\{1, 2\}$	$\{8, 10, 11\}$
$B_2$	$\{3, 4\}$	$\{5, 6\}$
$B_3$	$\{5\}$	$\{4, 6\}$
$B_4$	$\{6, 7\}$	$\{4, 5, 9\}$
$B_5$	$\{8, 9\}$	$\{2, 7, 11\}$
$B_6$	$\{10, 11\}$	$\{1, 2, 8\}$

然后迭代求 IN 和 OUT 集合, 如下

基本块	OUT <sup>0</sup>	IN <sup>1</sup>	OUT <sup>1</sup>	IN <sup>2</sup>	OUT <sup>2</sup>
$B_1$	$\emptyset$	$\emptyset$	$\{1, 2\}$	$\emptyset$	$\{1, 2\}$
$B_2$	$\emptyset$	$\{1, 2\}$	$\{1, 2, 3, 4\}$	$\{1, 2, 3, 4, 5, 8, 9\}$	$\{1, 2, 3, 4, 8, 9\}$
$B_3$	$\emptyset$	$\{1, 2, 3, 4\}$	$\{1, 2, 3, 5\}$	$\{1, 2, 3, 4, 6, 7, 8, 9\}$	$\{1, 2, 3, 5, 7, 8, 9\}$
$B_4$	$\emptyset$	$\{1, 2, 3, 5\}$	$\{1, 2, 3, 6, 7\}$	$\{1, 2, 3, 5, 7, 8, 9\}$	$\{1, 2, 3, 6, 7, 8\}$
$B_5$	$\emptyset$	$\{1, 2, 3, 4, 5\}$	$\{1, 3, 4, 5, 8, 9\}$	$\{1, 2, 3, 4, 5, 7, 8, 9\}$	$\{1, 3, 4, 5, 8, 9\}$
$B_6$	$\emptyset$	$\{1, 3, 4, 5, 8, 9\}$	$\{3, 4, 5, 8, 9, 10, 11\}$	$\{1, 3, 4, 5, 8, 9\}$	$\{3, 4, 5, 9, 10, 11\}$
EXIT	$\emptyset$	$\{3, 4, 5, 8, 9, 10, 11\}$	$\{3, 4, 5, 8, 9, 10, 11\}$	$\{3, 4, 5, 9, 10, 11\}$	$\{3, 4, 5, 9, 10, 11\}$

最后两列即为最终得到的 IN 和 OUT 集合.

- (b) 全体表达式为  $U = \{1, 2, a+b, c-a, b+d, e+1, b*d, a-d\}$ , 先求  $e\_gen$  和  $e\_kill$  集合, 如下

基本块	$e\_gen$	$e\_kill$
$B_1$	$\{1, 2\}$	$\{a+b, c-a, b+d, b*d, a-d\}$
$B_2$	$\{a+b, c-a\}$	$\{c-a, b+d, b*d, a-d\}$
$B_3$	$\emptyset$	$\{b+d, b*d, a-d\}$
$B_4$	$\{a+b\}$	$\{b+d, e+1, b*d, a-d\}$
$B_5$	$\{c-a\}$	$\{a+b, b+d, e+1, b*d\}$
$B_6$	$\{a-d\}$	$\{a+b, c-a, b+d, b*d\}$

然后迭代求 IN 和 OUT 集合, 如下

基本块	OUT <sup>0</sup>	IN <sup>1</sup>	OUT <sup>1</sup>
$B_1$	$U$	$\emptyset$	$\{1, 2\}$
$B_2$	$U$	$\{1, 2\}$	$\{1, 2, a+b, c-a\}$
$B_3$	$U$	$\{1, 2, a+b, c-a\}$	$\{1, 2, a+b, c-a\}$
$B_4$	$U$	$\{1, 2, a+b, c-a\}$	$\{1, 2, a+b, c-a\}$
$B_5$	$U$	$\{1, 2, a+b, c-a\}$	$\{1, 2, c-a\}$
$B_6$	$U$	$\{1, 2, c-a\}$	$\{1, 2, a-d\}$
EXIT	$U$	$\{1, 2, a-d\}$	$\{1, 2, a-d\}$

(c) 全体变量集合为  $\{a, b, c, d, e\}$ , 先求  $use$  和  $def$  集合, 如下

基本块	$use$	$def$
$B_1$	$\emptyset$	$\{a, b\}$
$B_2$	$\{a, b\}$	$\{c, d\}$
$B_3$	$\{b, d\}$	$\emptyset$
$B_4$	$\{a, b, e\}$	$\{d\}$
$B_5$	$\{a, b, c\}$	$\{e\}$
$B_6$	$\{b, d\}$	$\{a\}$

然后迭代求 IN 和 OUT 集合, 计算次序为  $B_6 \rightarrow B_5 \rightarrow B_3 \rightarrow B_4 \rightarrow B_2 \rightarrow B_1$ , 如下

基本块	IN <sup>0</sup>	OUT <sup>1</sup>	IN <sup>1</sup>	OUT <sup>2</sup>	IN <sup>2</sup>	OUT <sup>2</sup>	IN <sup>2</sup>
ENTER	$\emptyset$	$\emptyset$	$\emptyset$	$\{e\}$	$\{e\}$	$\{e\}$	$\{e\}$
$B_1$	$\emptyset$	$\{a, b\}$	$\emptyset$	$\{a, b, e\}$	$\{e\}$	$\{a, b, e\}$	$\{e\}$
$B_2$	$\emptyset$	$\{a, b, c, d\}$	$\{a, b\}$	$\{a, b, c, d, e\}$	$\{a, b, e\}$	$\{a, b, c, d, e\}$	$\{a, b, e\}$
$B_3$	$\emptyset$	$\{a, b, c, d\}$	$\{a, b, c, d\}$	$\{a, b, c, d, e\}$	$\{a, b, c, d, e\}$	$\{a, b, c, d, e\}$	$\{a, b, c, d, e\}$
$B_4$	$\emptyset$	$\{a, b, c, d\}$	$\{a, b, c, e\}$	$\{a, b, c, d, e\}$	$\{a, b, c, e\}$	$\{a, b, c, d, e\}$	$\{a, b, c, e\}$
$B_5$	$\emptyset$	$\{b, d\}$	$\{a, b, c, d\}$	$\{a, b, d\}$	$\{a, b, c, d\}$	$\{a, b, d, e\}$	$\{a, b, c, d\}$
$B_6$	$\emptyset$	$\emptyset$	$\{b, d\}$	$\emptyset$	$\{b, d\}$	$\emptyset$	$\{b, d\}$

最后两列即为最终得到的 IN 和 OUT 集合.

**习题 9.22.** 请利用代码优化的思想 (代码外提和强度削弱等), 改写下面 C 语言程序中的循环, 得到优化后的 C 语言程序.

```
main() {
    int i, j;
    int r[20][10];

    for (i = 0; i < 20; i++) {
        for (j = 0; j < 10; j++) {
            r[i][j] = 10 * i * j;
        }
    }
}
```

```
    }  
  }  
}
```

**解：**内循环中的  $i * 10$  是循环不变计算，可以提至外循环中。可以用加法代替乘法计算，因为每次执行外循环  $i * 10$  都只增加了 10，而每次执行内循环  $10 * i * j$  都只增加了  $10 * i$ 。同时可以用数组元素地址代替去下标运算，因为每次执行内循环  $r[i][j]$  都是数组的下一个元素。综合起来，优化后的 C 语言程序为

```
main() {  
    int i, j, m, n;  
    int *ptr;  
    int r[20][10];  
    ptr = &r[0][0];  
    m = 0; // m = 10 * i  
  
    for (i = 0; i < 20; i++) {  
        n = 0; // n = 10 * i * j  
        for (j = 0; j < 10; j++) {  
            *ptr = n;  
            ptr++;  
            n = n + m;  
        }  
        m = m + 10;  
    }  
}
```