

HW6

PB21111686_赵卓

7.2.(c)

- 三地址码如下：

```
t1 = 0
t2 = c /* base_a - 84 */
L0:
if t1 <= 10 goto L1
else goto L2
L1:
t3 = t1 * w /* 可取w = 4 */
t2[t3] = 0
/* 原c代码死循环，加一句t1 = t1 + 1 似乎更合理*/
goto L0
L2:
```

7.5

- 修改如下：

$P \rightarrow D; SP.offset = 0$

$D \rightarrow D; D$

$D \rightarrow id :$

$T \rightarrow enter(id.lexeme, T.type, P.offset), D.offset = D.offset + T.width$

$T \rightarrow integer \ T.type = integer, T.width = 4$

$T \rightarrow real \ T.type = real, T.width = 8$

$T \rightarrow$

$array[num] \ of \ T1 \ T.type = array(num.val, T1.type), num.val * T1.width$

$T \rightarrow T1 \ t.type = pointer(T1.type); T.width = 4$

7.12

- 三地址码如下：

```

t1 = x * 5
t1 = t1 + y
t2 = c /* c = base_A - 84 */
t3 = t1 * w /* 可取w = 4 */
t2[t3] = z

```

8.1.(e)

- 机器码如下：

```

MOV #a , R0
MOV #b , R1
ADD #c , R1
DIV R1 , R0
MOV R0 , M
MOV #d , R0
MOV #e , R1
ADD #f , R1
MUL R1 , R0
MOV M , R1
SUB R0 , R1
MOV R1 , M

```

8.2.(e)

- 机器码如下：

```

MOV #b , R0
MOV #c , R1
ADD R0 , R1
MOV R1 , SP
MOV #a , R0
DIV SP , R0
MOV R0 , 4(SP)
MOV #e , R1
ADD #f , R1
MOV R1 , 8(SP)
MOV #d , R0
MUL 8(SP) , R0
MOV 4(SP) , R1
SUB R0 , R1
MOV R1 , 12(SP)

```

8.6

- 原因如下：

- *x86*:

先处理 $(++i) + (++i)$ ，由于 $()$ 优先级先进行两次自增使得 $i = 2$ ，算得该式值为4；

然后处理后一个 $(++i)$ ， i 再次自增为3，结果为 $4 + 3 = 7$ 。

- *SPARC*:

依次处理三个 $(++i)$ ，因此 i 依次递增为1，2，3，结果为 $1 + 2 + 3 = 6$ 。