

练习 6.1.1: 练习 6.2.1: 假定图 6-26 中的函数 `widen` 可以处理图 6-25a 的层次结构中的所有类型，翻译下列表达式。假定 `d` 是 `char` 型，`s` 和 `t` 是 `short` 型，`x` 是 `float` 型。

1) `x = s + d`

2) `x = (s + d) * (t + x)`

```
1)
x = s + d
t1 = (int)s
t2 = (int)d
t3 = t1 + t2
x = (float)t3
2)
t1 = (int)s
t2 = (int)d
t3 = t1 + t2
t4 = (float)t
t5 = t4 + x
t6 = (float)t3
x = t6 * t5
```

练习 6.2.2: 在图 6-36 的语法制导定义中添加处理下列控制流构造的规则:

1) $S \rightarrow \text{repeat } S1 \text{ until } B$, 当 `B` 为真时结束循环

2) $S \rightarrow \text{for } (S1 ; B ; S2) S3$

```
1)
S1.next = newlabel()
B.true = S.next
B.false = newlabel()
S.code = label(B.false) || S1.code || label(S1.next) || B.code
2)
S1.next = newlabel()
B.true = newlabel()
B.false = S.next
S2.next = S1.next
S3.next = newlabel()
S.code = S1.code || label(S1.next) || B.code
|| label(B.true) || S3.code || label(S3.next) || S2.code || gen('goto', S1.next)
```

练习 7.1.1: 考虑 C 语言的函数 `f` 和 `g`: 按照图 7-7 的约定，不考虑编译器优化，讨论当 `f` 调用 `g` 而 `g` 即将返回时运行时栈的状态，其中 `f` 的参数 `a = 3`。只需要讨论返回值、参数、控制链和代码中体现的局部数据。

指出:

1) 哪个函数在栈中为各个元素创建了所使用的空间?

2) 哪个函数写入了各个元素的值? 参数、返回值和局部变量的值是什么?

3) 这些元素属于哪个活动记录?

```

int g(int *);
int f(int a) {
    int i = a + 2;
    return g(&i);
}
int g(int *b) {
    int j = *b;
    return j + 2;
}

```

	符号	创建者	写入者	值	谁的活动记录
1	int a	f 的调用者	f 的调用者	3	f
2	f 的返回值	f 的调用者	f		f
3	f 的控制链	f 的调用者	f 的调用者		f
4	int i	f	f	5	f
5	int *b	f	f	i 的地址	g
6	g 的返回值	f	g	7	g
7	g 的控制链	f	f		g
8	int j	g	g	5	g

练习 7.1.2: 考虑下面的 Fibonacci 函数:

嵌套在 fib0 中的是 fib1, 它假设 $n \geq 2$ 并计算第 n 个 Fibonacci 数。嵌套在 fib1 中的是 fib2, 它假设 $n \geq 4$ 。请注意, fib1 和 fib2 都不需要检查基本情况。我们考虑从对 main 的调用开始, 直到 (对 fib0(1) 的) 第一次调用即将返回的时段。

3) 请描述出当时的活动记录栈, 并给出栈中的各个活动记录的访问链。

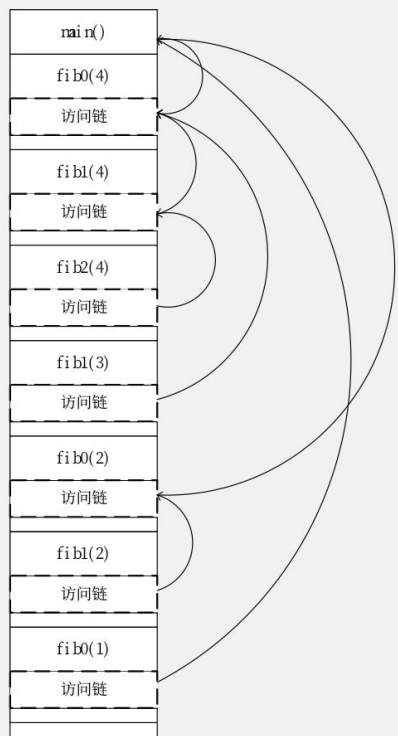
4) 假设我们使用 display 表来实现下图中的函数。请给出 fib0(1) 的第一次调用即将返回时的 display 表。同时指明那时在栈中的各个活动记录中保存的 display 表条目。

```

fun main() {
  let
    fun fib0(n) =
      let
        fun fib1(n) =
          let
            fun fib2(n) = fib1(n-1) + fib1(n-2)
          in
            if n >= 4 then fib2(n)
            else fib0(n-1) + fib0(n-2)
          in
            if n >= 2 then fib1(n)
          end
        end
      in
        fib0(4)
      end;
}

```

1)



2)

