```
declare void @llvm.va_end(ptr) #1

; Function Attrs: noinline nounwind optnone ssp uwtable(sync)
define i32 @Fibonacci(i32 noundef %0) #0 !dbg !21 {    ; int Fibonacci(int %0) -> int Fibonacci(int n)
  ; %1 = BB1
  %2 = alloca i32, align 4                             ; int %2
  %3 = alloca i32, align 4                             ; int %3
  store i32 %0, ptr %3, align 4                        ; %3 = %0 -> %3 = n
  %4 = load i32, ptr %3, align 4, !dbg !22             ; %4 = %3 -> %4 = n
  %5 = icmp eq i32 %4, 0, !dbg !23                     ; %5 = true if n == 0 else false
  br i1 %5, label %6, label %8, !dbg !22               ; jump based on %5 to label 6 or label 8

6:                                                     ; preds = %1
  ; %6 = BB2
  %7 = call i32 (ptr, ...) @printf(ptr noundef @.str), !dbg !24    ;%7 = printf(.str) -> %7 = printf("f(0) = 0")
  store i32 0, ptr %2, align 4, !dbg !25               ; %2 = 0
  br label %27, !dbg !25                               ; unconditional jump to label 27

8:                                                     ; preds = %1
  ; %8 = BB3
  %9 = load i32, ptr %3, align 4, !dbg !26             ; %9 = %3 -> %9 = n
  %10 = icmp eq i32 %9, 1, !dbg !27                    ; %10 = true if n == 1 else false
  br i1 %10, label %11, label %13, !dbg !26            ; jump based on %10 to label 11 or label 13

11:                                                    ; preds = %8
  ; %11 = BB4
  %12 = call i32 (ptr, ...) @printf(ptr noundef @.str.1), !dbg !28    ; %12 = printf(.str.1) -> %12 = printf("f(1) = 1")
  store i32 1, ptr %2, align 4, !dbg !29               ; %2 = 1
  br label %27, !dbg !29

13:                                                    ; preds = %8
  ; %13 = BB5
  %14 = load i32, ptr %3, align 4, !dbg !30            ; %14 = %3 -> %14 = n
  %15 = load i32, ptr %3, align 4, !dbg !31            ; %15 = %3 -> %15 = n
  %16 = sub nsw i32 %15, 1, !dbg !32                   ; %16 = %15 - 1 -> %16 = n - 1
  %17 = load i32, ptr %3, align 4, !dbg !33            ; %17 = %3 -> %17 = n
  %18 = sub nsw i32 %17, 2, !dbg !34                   ; %18 = %17 - 2 -> %18 = n - 2
  %19 = call i32 (ptr, ...) @printf(ptr noundef @.str.2, i32 noundef %14, i32 noundef %16, i32 noundef %18), !dbg !35   ; %19 = printf(.str.2, %14, %16, %18) -> %19 = printf("f(%d) = f(%d) + f(%d)", %14, %16, %18)
  %20 = load i32, ptr %3, align 4, !dbg !36            ; %20 = %3 -> %20 = n
  %21 = sub nsw i32 %20, 1, !dbg !37                   ; %21 = %20 - 1 -> %21 = n - 1
  %22 = call i32 @Fibonacci(i32 noundef %21), !dbg !38       ; %22 = Fibonacci(%21) -> %22 = Fibonacci(n - 1)
  %23 = load i32, ptr %3, align 4, !dbg !39            ; %23 = %3 -> %23 = n
  %24 = sub nsw i32 %23, 2, !dbg !40                   ; %24 = %23 - 2 -> %24 = n - 2
  %25 = call i32 @Fibonacci(i32 noundef %24), !dbg !41       ; %25 = Fibonacci(%24) -> %25 = Fibonacci(n - 2)
  %26 = add nsw i32 %22, %25, !dbg !42                 ; %26 = %22 + %25 -> %26 = Fibonacci(n - 1) + Fibonacci(n - 2)
  store i32 %26, ptr %2, align 4, !dbg !43             ; %2 = %26
  br label %27, !dbg !43                               ; unconditional jump to label 27

27:                                                    ; preds = %13, %11, %6
  ; %27 = BB6
  %28 = load i32, ptr %2, align 4, !dbg !44            ; %28 = %2
  ret i32 %28, !dbg !44                                ; return %28
}

attributes #0 = { noinline nounwind optnone ssp uwtable(sync) "frame-pointer"="non-leaf" "min-legal-vector-width"="0" "no-trapping-math"="true" "probe-stack"="__chkstk_darwin" "stack-protector-buffer-size"="8" "target-cpu"="
attributes #1 = { nocallback nofree nosync nounwind willreturn }
```