

```
entry:
  %call = tail call i32 @initialise_arrays(i8* getelementptr inbounds ([5 x
... i8], [5 x i8]* @ _func_.s000, i64 0, i64 0)) #11
  %t1 = getelementptr inbounds %struct.args_t, %struct.args_t* %func_args, i64
... 0, i32 0
  %call1 = tail call i32 @gettimeofday(%struct.timeval* %t1, i8* null) #11
  br label %for.cond2.preheader
```

```
for.cond2.preheader:
  %nl.022 = phi i32 [ 0, %entry ], [ %inc10, %for.cond.cleanup4 ]
  br label %for.body5
```

```
for.body5:
  %indvars.iv = phi i64 [ 0, %for.cond2.preheader ], [ %indvars.iv.next.7,
... %for.body5 ]
  %arrayidx = getelementptr inbounds [32000 x float], [32000 x float]* @b, i64
... 0, i64 %indvars.iv
  %0 = load float, float* %arrayidx, align 32, !tbaa !4
  %add = fadd float %0, 1.000000e+00
  %arrayidx7 = getelementptr inbounds [32000 x float], [32000 x float]* @a,
... i64 0, i64 %indvars.iv
  store float %add, float* %arrayidx7, align 32, !tbaa !4
  %indvars.iv.next = or i64 %indvars.iv, 1
  %arrayidx.1 = getelementptr inbounds [32000 x float], [32000 x float]* @b,
... i64 0, i64 %indvars.iv.next
  %1 = load float, float* %arrayidx.1, align 4, !tbaa !4
  %add.1 = fadd float %1, 1.000000e+00
  %arrayidx7.1 = getelementptr inbounds [32000 x float], [32000 x float]* @a,
... i64 0, i64 %indvars.iv.next
  store float %add.1, float* %arrayidx7.1, align 4, !tbaa !4
  %indvars.iv.next.1 = or i64 %indvars.iv, 2
  %arrayidx.2 = getelementptr inbounds [32000 x float], [32000 x float]* @b,
... i64 0, i64 %indvars.iv.next.1
  %2 = load float, float* %arrayidx.2, align 8, !tbaa !4
  %add.2 = fadd float %2, 1.000000e+00
  %arrayidx7.2 = getelementptr inbounds [32000 x float], [32000 x float]* @a,
... i64 0, i64 %indvars.iv.next.1
  store float %add.2, float* %arrayidx7.2, align 8, !tbaa !4
  %indvars.iv.next.2 = or i64 %indvars.iv, 3
  %arrayidx.3 = getelementptr inbounds [32000 x float], [32000 x float]* @b,
... i64 0, i64 %indvars.iv.next.2
  %3 = load float, float* %arrayidx.3, align 4, !tbaa !4
  %add.3 = fadd float %3, 1.000000e+00
  %arrayidx7.3 = getelementptr inbounds [32000 x float], [32000 x float]* @a,
... i64 0, i64 %indvars.iv.next.2
  store float %add.3, float* %arrayidx7.3, align 4, !tbaa !4
  %indvars.iv.next.3 = or i64 %indvars.iv, 4
  %arrayidx.4 = getelementptr inbounds [32000 x float], [32000 x float]* @b,
... i64 0, i64 %indvars.iv.next.3
  %4 = load float, float* %arrayidx.4, align 16, !tbaa !4
  %add.4 = fadd float %4, 1.000000e+00
  %arrayidx7.4 = getelementptr inbounds [32000 x float], [32000 x float]* @a,
... i64 0, i64 %indvars.iv.next.3
  store float %add.4, float* %arrayidx7.4, align 16, !tbaa !4
  %indvars.iv.next.4 = or i64 %indvars.iv, 5
  %arrayidx.5 = getelementptr inbounds [32000 x float], [32000 x float]* @b,
... i64 0, i64 %indvars.iv.next.4
  %5 = load float, float* %arrayidx.5, align 4, !tbaa !4
  %add.5 = fadd float %5, 1.000000e+00
  %arrayidx7.5 = getelementptr inbounds [32000 x float], [32000 x float]* @a,
... i64 0, i64 %indvars.iv.next.4
  store float %add.5, float* %arrayidx7.5, align 4, !tbaa !4
  %indvars.iv.next.5 = or i64 %indvars.iv, 6
  %arrayidx.6 = getelementptr inbounds [32000 x float], [32000 x float]* @b,
... i64 0, i64 %indvars.iv.next.5
  %6 = load float, float* %arrayidx.6, align 8, !tbaa !4
  %add.6 = fadd float %6, 1.000000e+00
  %arrayidx7.6 = getelementptr inbounds [32000 x float], [32000 x float]* @a,
... i64 0, i64 %indvars.iv.next.5
  store float %add.6, float* %arrayidx7.6, align 8, !tbaa !4
  %indvars.iv.next.6 = or i64 %indvars.iv, 7
  %arrayidx.7 = getelementptr inbounds [32000 x float], [32000 x float]* @b,
... i64 0, i64 %indvars.iv.next.6
  %7 = load float, float* %arrayidx.7, align 4, !tbaa !4
  %add.7 = fadd float %7, 1.000000e+00
  %arrayidx7.7 = getelementptr inbounds [32000 x float], [32000 x float]* @a,
... i64 0, i64 %indvars.iv.next.6
  store float %add.7, float* %arrayidx7.7, align 4, !tbaa !4
  %indvars.iv.next.7 = add nuw nsw i64 %indvars.iv, 8
  %exitcond.7.not = icmp eq i64 %indvars.iv.next.7, 32000
  br i1 %exitcond.7.not, label %for.cond.cleanup4, label %for.body5,
... !llvm.loop !8
```

T

F

```
for.cond.cleanup4:
  %call8 = tail call i32 @dummy(float* getelementptr inbounds ([32000 x
... float], [32000 x float]* @a, i64 0, i64 0), float* getelementptr inbounds
... ([32000 x float], [32000 x float]* @b, i64 0, i64 0), float* getelementptr
... inbounds ([32000 x float], [32000 x float]* @c, i64 0, i64 0), float*
... getelementptr inbounds ([32000 x float], [32000 x float]* @d, i64 0, i64 0),
... float* getelementptr inbounds ([32000 x float], [32000 x float]* @e, i64 0,
... i64 0), [256 x float]* getelementptr inbounds ([256 x [256 x float]], [256 x
... [256 x float]]* @aa, i64 0, i64 0), [256 x float]* getelementptr inbounds
... ([256 x [256 x float]], [256 x [256 x float]]* @bb, i64 0, i64 0), [256 x
... float]* getelementptr inbounds ([256 x [256 x float]], [256 x [256 x float]]*
... @cc, i64 0, i64 0), float 0.000000e+00) #11
  %inc10 = add nuw nsw i32 %nl.022, 1
  %exitcond.not = icmp eq i32 %inc10, 200000
  br i1 %exitcond.not, label %for.cond.cleanup, label %for.cond2.preheader,
... !llvm.loop !2
```

T

F

```
for.cond.cleanup:
  %t2 = getelementptr inbounds %struct.args_t, %struct.args_t* %func_args, i64
... 0, i32 1
  %call12 = tail call i32 @gettimeofday(%struct.timeval* nonnull %t2, i8*
... null) #11
  %call13 = tail call float @calc_checksum(i8* getelementptr inbounds ([5 x
... i8], [5 x i8]* @ _func_.s000, i64 0, i64 0)) #11
  ret float %call13
```

CFG for 's000' function