```
entry:
                                                  %call = tail call i32 @initialise arrays(i8* getelementptr inbounds ([5 x
                                                 ... i8], [5 x i8]* @ func .s271, i64 0, i64 0)) #6
                                                  %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) #6
                                                  br label %for.cond2.preheader
                                                          for.cond2.preheader:
                                                          %nl.029 = phi i32 [ 0, %entry ], [ %inc15, %for.cond.cleanup4 ]
                                                          br label %for.body5
                           for.bodv5:
                            %indvars.iv = phi i64 [ 0, %for.cond2.preheader ], [ %indvars.iv.next,
                            %arrayidx = getelementptr inbounds [32000 x float], [32000 x float]* @b, i64
                            .. 0, i64 %indvars.iv
                            %0 = load float, float* %arrayidx, align 4, !tbaa !4
                            %cmp6 = fcmp ogt float %0, 0.000000e+00
                            br i1 %cmp6, label %if.then, label %for.inc
                                                                                        F
if.then:
%arrayidx10 = getelementptr inbounds [32000 x float], [32000 x float]* @c,
... i64 0, i64 %indvars.iv
%1 = load float, float* %arrayidx10, align 4, !tbaa !4
%mul = fmul float %0, %1
%arrayidx12 = getelementptr inbounds [32000 x float], [32000 x float]* @a,
... i64 0, i64 %indvars.iv
%2 = load float, float* %arrayidx12, align 4, !tbaa !4
%add = fadd float %2, %mul
store float %add, float* %arrayidx12, align 4, !tbaa !4
br label %for.inc
                           for.inc:
                           %indvars.iv.next = add nuw nsw i64 %indvars.iv, 1
                           %exitcond.not = icmp eq i64 %indvars.iv.next, 32000
                           br i1 %exitcond.not, label %for.cond.cleanup4, label %for.body5, !llvm.loop
                           ... !8
                                             Τ
                                                                                      F
                             for.cond.cleanup4:
                              %call13 = 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]* getelement ptr 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) #6
                             %inc15 = add nuw nsw i32 %nl.029, 1
                              %exitcond30.not = icmp eq i32 %inc15, 400000
                              br i1 %exitcond30.not, label %for.cond.cleanup, label %for.cond2.preheader,
                             ...!llvm.loop!2
                                                                                          F
       for.cond.cleanup:
       %t2 = getelementptr inbounds %struct.args t, %struct.args t* %func args, i64
       ... 0, i32 1
       %call17 = tail call i32 @gettimeofday(%struct.timeval* nonnull %t2, i8*
       ... null) #6
       %call18 = tail call float @calc checksum(i8* getelementptr inbounds ([5 x
```

... i8], [5 x i8]\* @ func .s271, i64 0, i64 0)) #6

ret float %call18