

## Malloc to alloca

### Pseudocode

; Find all malloc variables

vector <Instruction> mall;

for (BasicBlock BB : F)

    for (Instruction I : BB) {

        if I matches malloc

            add to mall

    }

; Check whether the instruction is freed before end of function, if not, remove

bool flag = true

foreach m in mall {

    for (BasicBlock BB : F)

        for (Instruction I : BB) {

            if I matches free and m in arguments

                flag = false

        }

    if (flag)

        remove m from mall

}

; for each instruction in mall change malloc to alloca

foreach m in mall {

    change malloc to alloca

}

## Example

; Before optimization

```
define i32 @f(i32 %x, i32 %y) {  
    %malloc_var = call i8* @malloc(i32 4)  
  
    ...  
  
    call void @free(i8* %malloc_var)  
  
    ret 0  
}
```

; After optimization

```
define i32 @f(i32 %x, i32 %y) {  
    %alloca_var = alloca %i32  
  
    ...  
  
    ret 0  
}
```