

```

#include <stdio.h>
#include <string.h>

union color {
    struct {
        char r;
        char g;
        char b;
    } rgb;
    struct {
        float h;
        float s;
        float l;
    } hsl;
};

const union color pink = {.hsl = {.h = 330.0f, .s = 0.75f, .l = 0.5f}};
const union color black = {.rgb = {.r = 0, .g = 0, .b = 0}};

void lighten(const union color *, union color *, float);

int main(int argc, char *argv[])
{
    static union color lighter_pink;
    lighten(&pink, &lighter_pink, 0.2f);

    printf(
        "Lighter pink has H%f S%f L%f\n",
        lighter_pink.hsl.h,
        lighter_pink.hsl.s,
        lighter_pink.hsl.l
    );
    return 0;
}

void lighten(const union color *from, union color *to, float percentage)
{
    memcpy(to, from, sizeof(union color));
    to->hsl.l += percentage;
    to->hsl.l = to->hsl.l < 0.0f
        ? 0.0f
        : to->hsl.l > 1.0f
        ? 1.0f
        : to->hsl.l;
}

```

```

.file      "hello.c"
.text
.globl     lighten
.type      lighten, @function
lighten:
.LFB12:
.cfi_startproc
movq      (%rdi), %rax
movq      %rax, (%rsi)
movl      8(%rdi), %eax
movl      %eax, 8(%rsi)
addss     8(%rsi), %xmm0
movss     %xmm0, 8(%rsi)
pxor      %xmm1, %xmm1
comiss     %xmm0, %xmm1
ja        .L4
comiss     .LC1(%rip), %xmm0
jbe       .L2
movss     .LC1(%rip), %xmm0
jmp       .L2
.L4:
pxor      %xmm0, %xmm0
.L2:
movss     %xmm0, 8(%rsi)
ret
.cfi_endproc
.LFE12:
.size     lighten, .-lighten
.section   .rodata.str1.1, "aMS",@progbits,1
.LC3:
.string    "Ligher pink has H%f S%f L%f\n"
.text
.globl     main
.type      main, @function
main:
.LFB11:
.cfi_startproc
subq      $8, %rsp
.cfi_def_cfa_offset 16
movss     .LC2(%rip), %xmm0
leaq      lighter_pink.0(%rip), %rsi
leaq      pink(%rip), %rdi
call      lighten
pxor      %xmm0, %xmm0
cvtss2sd   lighter_pink.0(%rip), %xmm0
pxor      %xmm2, %xmm2
cvtss2sd   8+lighter_pink.0(%rip), %xmm2
pxor      %xmm1, %xmm1
cvtss2sd   4+lighter_pink.0(%rip), %xmm1
leaq      .LC3(%rip), %rdi
movl      $3, %eax

```

```

    call    printf@PLT
    movl    $0, %eax
    addq    $8, %rsp
    .cfi_def_cfa_offset 8
    ret
    .cfi_endproc
.LFE11:
    .size    main, .-main
    .local   lighter_pink.0
    .comm    lighter_pink.0,12,8
    .globl   black
    .section .rodata
    .align 8
    .type    black, @object
    .size    black, 12
black:
    .zero    12
    .globl   pink
    .align 8
    .type    pink, @object
    .size    pink, 12
pink:
    .long    1134886912
    .long    1061158912
    .long    1056964608
    .section .rodata.cst4,"aM",@progbits,4
    .align 4
.LC1:
    .long    1065353216
    .align 4
.LC2:
    .long    1045220557
    .ident    "GCC: (Debian 10.2.1-6) 10.2.1 20210110"
    .section .note.GNU-stack,"",@progbits

```

C 소스 파일인 hello.c 파일과 컴파일 결과물인 hello.s 파일을 비교한다.

먼저, 최상단의 #include 구문은 컴파일 시 사용 가능한 symbol에 대한 정보를 제공하므로 출력되는 어셈블리 코드에 영향은 미치지 않는다. 다음으로 눈여겨볼 점은 union color이다. 이 공용체는 struct { char; char; char; }와 struct { float; float; float; }의 공용체로, sizeof(union color)는 sizeof(char) * 3 < sizeof(float) * 3 이므로 메모리 내 위치를 고려했을 때 12이다.

한 가지 흥미로운 점은 const union color pink의 정의에서 찾아볼 수 있다. pink의 hsl 원도는 각각 330.0f, 0.75f, 0.5f로 초기화되는데, 어셈블리 상에서는 이 숫자가 1134886912, 1061158912, 1056964608으로 바뀌어 있다. 이는 IEEE-754 binary32 규격에 맞게, 각 byte를 비트 표현식으로 변환하는 것이다. const union color black의 경우 (R, G, B) 값이 모두 0이므로 .zero directive를 사용해서 rodata section에 초기화하는 것을 볼 수 있다.

lighten 함수에서는 memcpy 호출이 최적화되어 사라진 것을 제외하면 크기 다른 것은 없다. 다만 main 함수의 static union color lighter-pink 변수의 처리 방법이 인상적인데, 다른 지역 변수와 달리 스택에 저장하지 않고 함수 내부에 저장되는 것을 확인한 두 있다. 이는 C언어 명세 상 함수 내에서 선언된 정적 변수는 함수가 종료된 이후에도 상태를 저장해야 하기 때문이다. 만약 정적 변수를 다른 변수와 같이 스택에 할당하게 되면 ret 명령을 호출하는 시점에서 rsp 레지스터가 복귀되며 해당 정적 변수의 상태 유지를 보장할 수 없게 된다. 이에 컴파일러는 프로그램 실행 공간에 따른 함수 정적 변수를 저장한 공간을 할당하여 이러한 사태를 방지한다.

마지막으로, main 함수에서 printf 함수를 호출할 때, lighter-pink의 주소에서 오프셋은 직접 지정하여 변수 연산 레지스터에 집어넣는 것 또한 인상적이다. printf 역시 일반적인 범용 레지스터에서 값을 받지 않고, xmmN으로 명명된 xmm 레지스터에서 값을 받아온다. 이는 컴파일러가 해당 자료형의 모든 타입과 크기로 아키텍처에 있기에 가능한 작업으로, printf에 대한 특별한 최적화가 병행되었을 때 이러한 코드가 도출된다.