

No código C fornecido, o ponteiro `pc` é inicializado para apontar para o endereço da variável `c`. A instrução `pc++` incrementa o ponteiro pelo tamanho do tipo de dados para o qual ele aponta, que neste caso é um `float`. Como um `float` ocupa 4 bytes na memória em um sistema de 32 bits, o ponteiro `pc` será incrementado em 4 bytes. Portanto, se o endereço inicial de `c` for `0x65fe14`, após o incremento, o ponteiro `pc` apontará para o endereço `0x65fe18`. Isso ocorre porque adicionar 4 bytes ao valor hexadecimal `0x65fe14` resulta em `0x65fe18`.

O ponteiro `pc` estará apontado para o endereço `0x65fe18`. Isso porque, quando você incrementa um ponteiro, ele avança na memória de acordo com o tamanho do tipo de dado que ele aponta. No caso, `pc` é um ponteiro para `float`, que ocupa 4 bytes em memória. Então, ao fazer `pc++`, você está somando 4 ao endereço de `pc`, que era `0x65fe14`. Logo, o novo endereço de `pc` é $0x65fe14 + 4 = 0x65fe18$.