Some CP edge-case qns

~SiD

1

```c
#include <stdio.h>

int main() {
    float a = 0.1;
    if (a + a + a + a + a + a + a + a + a + a == 1.0)
        printf("Equal\n");
    else
        printf("Not Equal\n");
    return 0;
}
```

Ans: Not equal (Floating point rounding error, 0.1 is truncated and hence its stored value is slightly different from 0.1)

2

```c
#define x 5 + 5
int main() {
    printf("%d\n", x * x);
    return 0;
}
```

Ans: 35 (Direct substitution of x as 5+5 gives: 5 + 5 * 5 + 5)

3

```c
#include <stdio.h>
#define SQUARE(x) x * x

int main() {
    printf("%d\n", SQUARE(2 + 3));
    return 0;
}
```

Ans: 11: Similar to the question above. Don’t treat it as a function

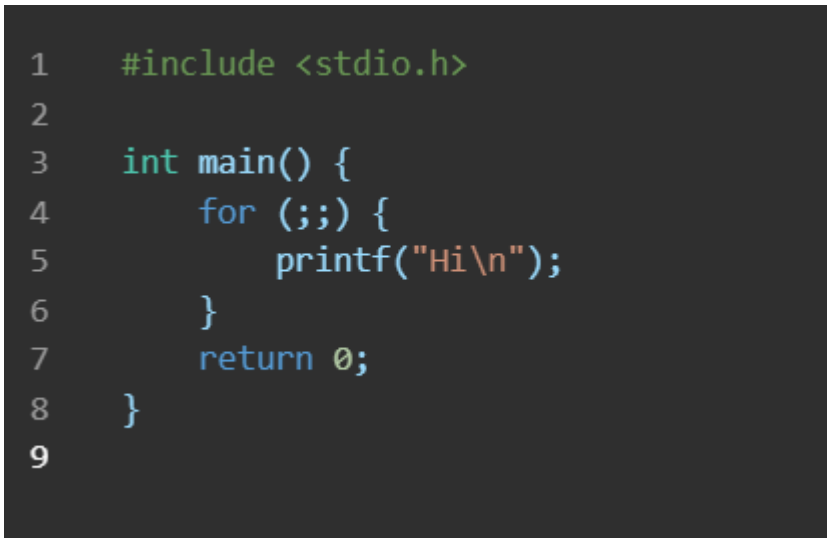
4

Ans: no
0.1 needs to be truncated (infinitely repeating binary representation)

a is first created as a double, truncated as a double and then its size is increased to that of a long double

b is created as a long double and then truncated
So their values won’t match

5



Ans: Infinite loop, when condition is not mentioned it is treated as true.