Order of Evaluation Solutions

Order of Evaluation in Older C++

- In C++14 and earlier, what will be the values of i and arr[i] after the following statement is executed?

```
int i = 0;
arr[i++] = i;            // May result in arr[0] == 0 or arr[0] == 1 or ...
```

- Explain your answr
  - Before C++17, the order of evaluation is not defined
  - The accesses to i can occur in any order
  - They can even interleave, causing data corruption

Function Arguments

- In C++14 and earlier, how would you expect the following function call to be performed?

```
// f1() could be called before f2(), or vice versa, or they could interleave
// x, y and z could be evaluated in any order
func(f1(x), f2(y, z));
```

- Explain your answer
  - Before C++17, the order of evaluation is not defined
  - f1() and f2() can be called in any order
  - Their arguments can be evaluated in any order