Consider the below two programs:

```
// Program 1
int main()
{
   int x;
   int x = 5;
   printf("%d", x);
   return 0;
}
```

Output in C:

```
redeclaration of ‘x’ with no linkage
```

```
// Program 2
int x;
int x = 5;

int main()
{
   printf("%d", x);
   return 0;
}
```

Output in C:

```
5
```

In C, the first program fails in compilation, but second program works fine. In C++, both programs fail in compilation.

C allows a global variable to be declared again when first declaration doesn’t initialize the variable.

The below program fails in both C also as the global variable is initialized in first declaration itself.

```
int x = 5;
int x = 10;

int main()
{
   printf("%d", x);
   return 0;
}
```

Output:

```
error: redefinition of ‘x’
```