

In C, the `void main()` function is often misunderstood. Let me clarify it for you:

1. **`int main()`:**

- This is the standard and recommended way to define the `main` function in C.
- It indicates that the `main` function takes no arguments.
- The return type is `int`, which means it should return an integer value.
- The purpose of returning an integer value from `main` is to provide an exit status to the operating system. Typically, a return value of 0 indicates successful execution, while other values indicate errors or abnormal termination¹.

2. **`void main()`:**

- Using `void main()` is **not** standard C.
- It indicates that the `main` function does not return any value.
- The C standard does not define the behavior of `void main()`, and its usage can lead to unpredictable program behavior.
- Some older compilers might accept it, but it's best to avoid this form.
- Stick to `int main()` or `int main(int argc, char **argv)` for portability and correctness^{1 3}.

In summary, always use `int main()` for your C programs. It ensures consistency, compatibility, and adherence to the language standard. 😊

Source(s)

1. [c - How does int main\(\) and void main\(\) work? - Stack Overflow](#)
2. [What Is A Void Main\(\) In The C Programming Language?](#)
3. [What is void main \(\) into main and what it does? - UrbanPro](#)
4. [Difference between void main and int main in C/C++](#)
5. [Difference between void main and int main in C/C++?](#)