In C++, both **function(\*&head)** and **function(\*head)** are function declarations that take a pointer to an object as an argument, but they have different semantics.

**function(\*head)** declares a function that takes a pointer to an object by value. This means that a copy of the pointer is passed to the function, and any changes made to the pointer within the function will not be reflected outside of the function.

On the other hand, **function(\*&head)** declares a function that takes a reference to a pointer to an object. This means that changes made to the pointer within the function will be reflected outside of the function. The ampersand & in the argument indicates that it is a reference parameter.

In other words, **function(\*&head)** allows the function to modify the pointer itself, such as changing what object it points to or setting it to null, whereas **function(\*head)** only allows the function to modify the object that the pointer points to.