

5/10/23

Q1) → Wild Pointers in C++

↳ A wild pointer is a pointer in C++ that is uninitialized or has been deleted. This means that the pointer does not point to a valid memory location, & accessing or dereferencing a wild pointer can result in undefined behavior.

Ex → `int *ptr;` → uninitialized
`cout << *ptr`

↳ Refer to a garbage value or throws a error

Uninitialize
(Declare)

Ex 2) → Deleting pointer

`int *ptr = new int;`

`delete ptr;` → delete

↳ `cout << *ptr << endl;`
 ↳ get an undefined behaviour

Ex 3) → Pointer to non-existing variable

`int *ptr = &x;`

`cout << *ptr;` → can crash the program

→ Note: ↳ Always initialize pointer
 ↳ Avoid deleting pointer

→ If you are deleting, then set it to nullptr to prevent it from becoming wild pointer

→ Always check pointer is pointing to a valid location