More on arrays and programming

We usually declare an array as int a[10]

The length of the array must be a constant that is know when your program is compiled (static)

We can overcome the limitation by allocating memory at runtime (dynamic) by using malloc()

Our declaration now becomes int \*a, indicating that a is a pointer to one or more ints. All we have done is declare the name, so we still have to allocate the actual memory.

After calling malloc(), we should always check the result, we can make our life easy by using assert()

Should check if there is remaining space.

Implementary a set using an array

We will need to keep track of 3 things:

1. The array “a”
2. The number of elements in the array “n”
3. The length of the array “m”

Logically, we should group these three things together into a single unit using structure.

An abstract data type (ADT) is a data type whose implementation is hidden from those functions that use it (a.k.a clients)

An ADT separates interface from implementation. Clients only see the interface: a set of functions

Interface🡪.h file

Implementation🡪.c file