**address** - a value that can be used to find an object in memory.

**address of: &** - creates a reference to the object or can be used to assign the address of already declared variable to the pointer

**allocation** - reserving of some memory to store an object

**cast** - an explicit type conversion

**container** - a data structure used to hold a series of elements

**contents of: \*** - used to derefence the pointer (get the value of an object pointer points to)

**deallocation** - an act of giving back (releasing) memory, that was allocated by the program

**delete** - deallocates memory used to store a single object

**delete[]** - deallocates memory used to store an array

**dereference** - get the value of the object that pointer points to

**destructor** - an operation that is implicitly invoked when the object is destroyed. Often, it releases resources

**free store** - a data segment which stores dynamically-allocated objects

**link** - an object that holds a pointer to the next element and the previous one

**list** - a sequence of objects that holds links to each other

**member access: –>** - used to access members and methods of the object that a pointer points to

**member destructor** - a destructor that is used to deallocate members of the class

**memory** - a hardware device that stores data in sequence of bytes

**memory leak** - losing control over the dynamically allocated memory, which can’t be used anymore by the computer to allocate it for another data.

**new** - an operator used to allocate memory for an object

**null pointer** - a pointer, that doesn’t refer to a valid object

**nullptr** - a keyword used to initialize pointer as a null pointer

**pointer** - a value that identifies a place of an object in memory

**range** - a sequence of elements that is defined by a start point and an end point.

**resource leak** - when some resource was given but wasn’t returned (released)

**subscripting** - an operation that retrieves an element with an index specified by the subcript within some sequence of elements

**subscript: [ ]** - an operator used to access elements of container

**this** - a keyword used to get a pointer to an object within its method

**type conversion**  - explicit change of object’s type

**virtual destructor**  - a destructor which is called when an object of derived class is accessed through a pointer to the base class.

**void\*** - a type of pointer that can’t be dereferenced. Usually used to represent a pointer that refers to an object of unknown type