# Lab 11: Dynamic array & pointers

For UNT CSCE1030 Summer 2021

TA Polina Nemkova

Poli.Nemkova@unt.edu

# What are Pointers?

- A **pointer** is a variable whose value is the address of another variable.

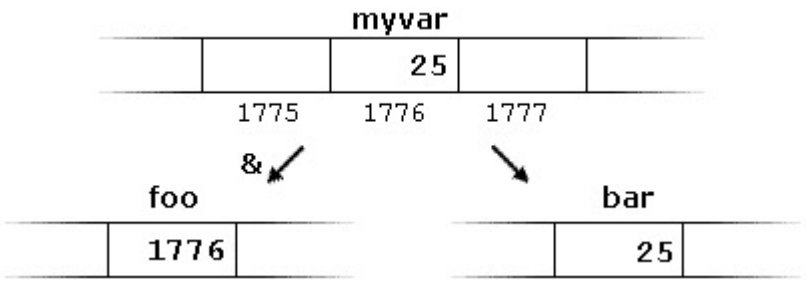
- Pointer declaration:

```
type *var-name;
```

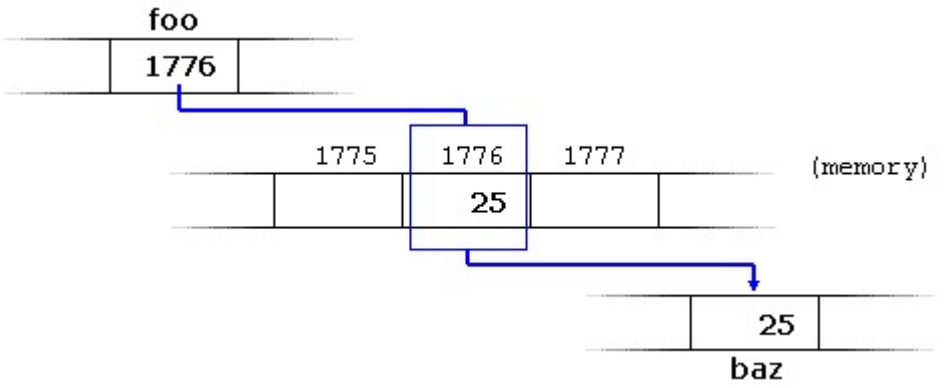
```
int *ip; // pointer to an integer
double *dp; // pointer to a double
float *fp; // pointer to a float
char *ch // pointer to character
```

# Working with pointers

**Address-of operator (&) – gives you address**




**Dereference operator (*) – gives you actual value**



# Dynamic arrays

- A dynamic array is quite similar to a regular array, but its size is modifiable during program runtime.
- Dynamic memory is allocated using operator **new**. Using **delete** we delete the dynamic array (or pointer)
- new is followed by a data type specifier and, if a sequence of more than one element is required, the number of these within brackets [].
- It returns a pointer to the beginning of the new block of memory allocated. Its syntax is:

```
pointer = new type
pointer = new type [number_of_elements]
```

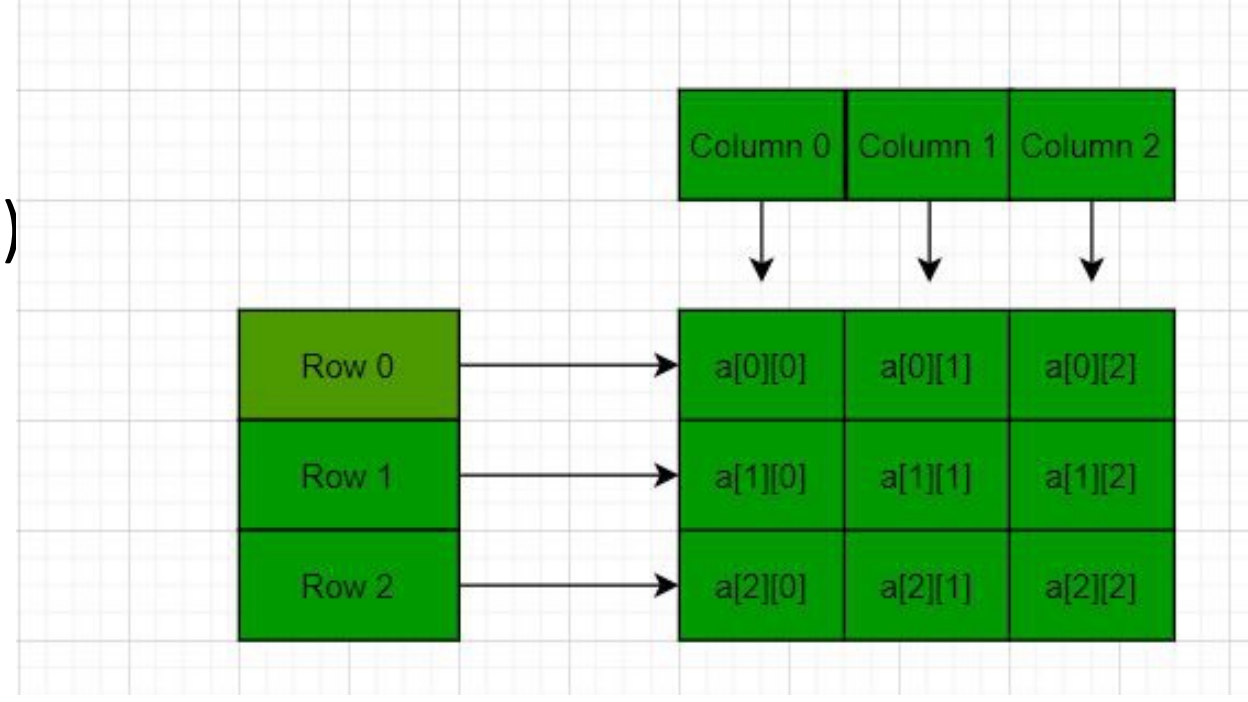
# 2-Dimensional dynamic array

- 2D arrays are arrays of single-dimensional arrays.
- Syntax
  Declaring:
  int** numlist = new int*[number_of_student];

- Deleting:
  for (int i=0;i<number_of_student;++i)

```
    delete [] numlist[i];
```

  delete [] numlist;





Tracing a pointer in C.