# BLG102E Introduction to Scientific & Engineering Computation

LABORATORY 13

Istanbul Technical University

June 11th, 2020

# swap() function

Write two swap() functions to replace integer variables.

- One with using pointers,
- The other one must be using references from C++
- Source file must end with .cpp instead of .c
- To compile:

```
g++ main.cpp -o main
```

Here are the declarations:

```
void swap(int *p1, int *p2);

void swapr(int &r1, int &r2);
```

```
int main()
{
   int x = 10;
   int y = 45;

   swap(&x, &y);

   printf("x = %d \n", x);
   printf("y = %d \n", y);

   swapr(x, y);

   printf("x = %d \n", x);
   printf("y = %d \n", y);

   return 0;
}
```

Test code

# class Name

Write a class to store, display and reverse a name in C++.

- Have a private `char *` variable to store a string (which is an address of a char array).
- Write the following class member functions:
    - Implement a constructor
    - Implement a destructor
    - Write a `display()` function
    - Write a `reverse()` function
- Source file must end with `.cpp` instead of `.c`
- To compile:

```
g++ main.cpp -o main
```

Here are the class and declarations:

```
class Name {
   char *mp;
public:
   Name(const char *p);
   ~Name();
   void display();
   void reverse();
};
```

```
int main()
{
   Name myname("nese");

   myname.display();
   myname.reverse();
   myname.display();

   return 0;
}
```

Test code

# class List

Write a class to store, display, push and pop nodes in a list in C++.

- Have a private `struct Node` which stores an integer type of data and an address of the next node.
- Write the following class member functions:
  - Implement a constructor
  - Implement a destructor
  - Write a `display()` function
  - Write a `push()` function
  - Write a `pop()` function

Here are the declarations:

```
class List {
   struct Node {
      //..
   };
public:
   List();
   ~List();
   void push(int val);
   void display();
   void pop();
};
```

```
int main()
{
   List mylist;

   for (int k = 0; k < 100; k++)
      mylist.push(k);

   mylist.display();

   for (int k = 0; k < 90; k++)
      mylist.pop();

   mylist.display();

   return 0;
}
```

Test code