## MEL

# Lecture 8 1D Arrays

and other disasters





#### WARSAW UNIVERSITY OF TECHNOLOGY Test is coming 5'th December

#### Know:

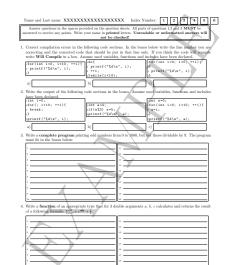
- Data types
- Functions
- I/O operations
- Branching (if, switch)
- Loops

#### Have:

- A Pen
- Student ID

#### Do not have:

- Notes
- Any electronic devices





**Today** 2D static arrays

Today:

- File I/O continue
- A program that preperas points for plotting a function
- A program that reads data from a file, and manipulates them.
- A program that generates random numbers and stores them in a file.
- A program that reads a file, and calculates an average
- A program that reads a file, sorts it and stores back

MEL

#### WARSAW UNIVERSITY OF TECHNOLOGY

▲□▶ ▲□▶ ▲□▶ ▲□▶ □ のQ@

**Files** 

```
FILE structure to handle files: FILE *fp;
```

```
To open a file use fopen():
FILE *fopen(const char *filename, const char *mode);
//e.g.:
fp=fopen("c:\\test.txt", "r");
```

```
To close a file use fclose():
int fclose(FILE *a_file);
//e.g.:
fclose(fp);
```



Files fopen modes

Depending on what we require the file to:

- r open for reading
- w open for writing (file need not exist)
- a open for appending (file need not exist)
- r+ open for reading and writing, start at beginning
- w+ open for reading and writing (overwrite file)
- a+ open for reading and writing (append if file exists)



▲ロト ▲帰ト ▲ヨト ▲ヨト 三日 - の々ぐ

#### **Files**

Reading and writing with fprintf, fscanf

```
Printing to file:
FILE *fp;
fp=fopen("c:\\test.txt", "w");
fprintf(fp, "Testing...\n");
...
fclose(fp);
Reading from file:
FILE *fp;
fp=fopen("c:\\test.txt", "w");
int a;
fcanf(fp, "¼d", &a);
...
fclose(fp);
```

### **Examples**

Use static arrays only.

- Write a program that writes to a file coordinates to plot f(x) = sin(x) for a range  $< 0, 2\pi >$
- Write program that reads points coordinates from a file and decides if those are in a circle of radius 1.
- Write a program that generates N random numbers and stores them to a file.
- Write a program that reads a data file, calculates an average value and finds the number of elements above, and below that average.
- **(5)** Write a program that reads values from a file, sorts them and stores them to a new file.
- 6 Example test questions