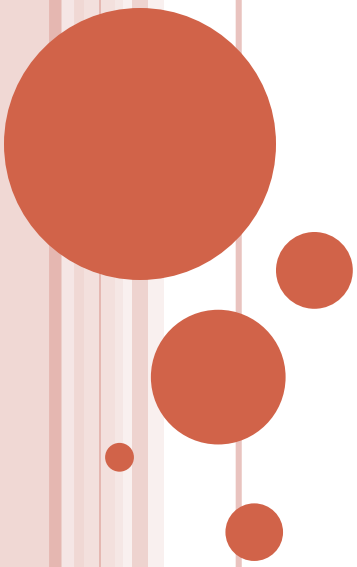

Chapter 9

Files

Standard Input and Output

- *printf(<control string>, <arguments>);*

- *scanf(<control string>, <arguments>);*

File Input and Output

- *fprintf(<file pointer>, <control string>, <arguments>);*
- *fscanf(<file pointer>, <control string>, <arguments>);*

File Pointers – Declaration

- *FILE *ifp;*

- Style Standards:
  - ifp = input file pointer
  - ofp = output file pointer

File Pointers – Initialization

- *fopen(<file name>, <mode>);*

- Modes:
  - “r” for read
  - “w” for write

Closing a File

- *fclose(<file pointer>);*

- *fclose(ifp);*

Example – Summation – input.txt

5

150

136

178

200

189

Practice

- Write a short program that reads numbers from a file and determines if each integer is even or odd. An example input is shown below – the first number indicates how many numbers are to be tested. The program should print "X is even." or "X is odd." for each number.

```
4
3
19
40
56
```

Example – Simple Output – output.txt

- *FILE *ofp = fopen("output.txt", "w");*

- *fprintf(ofp, "Hello World!");*

Example – Miles Ran – input.txt

5
5
3
7
2
8

- The first number is how many participants – the following integers represent the number of miles for a different participant.
- Our output file should read “Person X ran Y miles!”.

Practice

Suppose we actually wanted to print “Person X: “ followed by a * for each mile they ran. For example, the first person would have “Person 1: *****”. Adjust the code so that this output is printed.

Strings

Strings

- *char name[20];*

- A string called “name” of size 19 or less

Percent code %s

- *fscanf(ifp, "%s", &name);*
- *scanf("%s", &name);*

- *fprintf(ofp, "%s", name);*
- *printf("%s", name);*

Example – Miles Ran 2 – input.txt

5

George 5

Karla 3

Meredith 7

Jason 2

Aaron 8