

File Handling

Nalini Vasudevan
Columbia University

Files

- So far
 - input from the keyboard
 - `scanf`
 - output to the terminal
 - `printf`

Opening a file

```
FILE *fp;  
fp = fopen ("abc.txt", "r");
```

- `fp` is a file pointer
- Use `"w"` for write mode, `"a"` for append mode
- Prototype

```
FILE* fopen (char *name, char *mode);
```

Reading from a file

- Reading a character from a file

```
fgetc(fp);
```

- Returns an int (EOF or convert to char)

- Reading a string from a file

```
fgets(str, 256, fp);
```

Writing to a file

- Write a character to a file

```
fputc(c, fp);
```

- Write a string to a file

```
fputs(str, fp);
```

File Example

```
main()  
{  
    FILE *fpi, *fpo;  
    fpi = fopen("file.c", "r");  
    fpo = fopen("out.c", "w");  
    if (fpi == NULL || fpo == NULL)  
        printf("Error opening file\n");  
    else  
    {  
        filecopy (fpi, fpo);  
        fclose(fpi);  
        fclose(fpo);  
    }  
}
```

File Example

```
void filecopy(FILE *fpin, FILE *fpout)
{
    int c;
    while ((c = fgetc(fpin)) != EOF)
        fputc(c, fpout);
}
```