

Input/Output

# Keyboard Input

- scanf function
- from `<stdio.h>`
- `int x;`
- `scanf("%d", &x);`



# Examples

```
double y;
```

```
scanf("%lf", &y);
```

```
double *z = &y;
```

```
scanf("%lf", z);
```

# More on Pointers...

## **For arrays:**

```
int a[] = {1, 2, 3};
```

```
int *pointerToA = a; // a actually points to address!
```

## **For other types:**

```
int b = 4;
```

```
int *pointerToB = &b; // Must use address!
```

# Dereferencing Pointers

```
int a = 4;
```

```
int *b = &a;
```

```
int c = *b;
```

```
// c's value is 4
```

# Reading Strings

- Do not use scanf!
- Use scanner (download via activity)

# Command Line Input

- `$ prog 1 file.txt b`
- extra arguments when you run program
- `int argc, char **argv`
- stored as strings

A terminal window with a dark background and light-colored text. The title bar at the top shows window control icons and the text "root@linuxbox:~". The main content of the terminal shows the shell prompt "root@linuxbox:~\$" on the first line, with a blank line below it.

# File Input

```
FILE *fp = fopen("data","r")
if (fp == 0)
{
    fprintf(stderr,"file data could not be opened for reading\n");
    exit(1);
}

x = readInt(fp);
printf("the number read was %d\n",x);
fclose(fp);
```



# File Output

```
printf("hello, world!\n");  
fprintf(stdout, "hello, world!\n");
```

```
// delete the contents  
FILE *fp = fopen(fileName, "w")  
//check that fopen did not encounter a problem  
fclose(fp);
```

input.c