Introduction
○

Input Scenarios
○○○○

Using printf and scanf
○○

Interactive Tests
○

Introduction
●

Input Scenarios
○○○○

Using printf and scanf
○○

Interactive Tests
○

Input and Output

IO

Dr. Mattox Beckman

University of Illinois at Urbana-Champaign
Department of Computer Science

Your Objectives:

▶ Write input routines for three kinds of test inputs,

▶ use 'scanf' and 'printf' properly for various types of variables, and

▶ write code for interactive tests.

Introduction
○

Input Scenarios
●○○○

Using printf and scanf
○○

Interactive Tests
○

Introduction
○

Input Scenarios
○●○○

Using printf and scanf
○○

Interactive Tests
○

Explicit Test Count

▶ First line of input is the number of tests you will receive.

```c
#include <stdio.h>

int main() {
  int cases,x,y;
  scanf("%d",&cases);
  while (cases>0) {
    cases--;
    scanf("%d %d",&x,&y);
    printf("%d\n",x+y);
  }
}
```

Termination Marker

▶ The input itself will use a special value.

```c
#include <stdio.h>

int main() {
  int x,y;
  while (1) {
    scanf("%d %d",&x,&y);
    if (x==-1 && y==-1)
    break;
    printf("%d\n",x+y);
  }
}
```

Introduction
○

Input Scenarios
○○●○

Using `printf` and `scanf`
○○

Interactive Tests
○

Introduction
○

Input Scenarios
○○○●

Using `printf` and `scanf`
○○

Interactive Tests
○

Termination Marker, pt 2

```
0 #include <stdio.h>
1
2 int main() {
3   int x,y;
4   while (scanf("%d %d",&x,&y) && x != -1 && y != -1) {
5     if (x==-1 && y==-1)
6     break;
7     printf("%d\n",x+y);
8   }
9 }
```

End of File

▶ Use EOF explicitly.

```
0 #include <stdio.h>
1
2 int main() {
3   int x,y;
4   while (scanf("%d %d",&x,&y) != EOF) {
5     if (x==-1 && y==-1)
6     break;
7     printf("%d\n",x+y);
8   }
9 }
```

Introduction
○

Input Scenarios
○○○○

Using `printf` and `scanf`
●○

Interactive Tests
○

Introduction
○

Input Scenarios
○○○○

Using `printf` and `scanf`
○●

Interactive Tests
○

Why `scanf` and `printf`?

▶ There are problems that TLE if you use `cin` and `cout`.

▶ `scanf` has some regular-expression like features that can be useful.

Code	Meaning
%d	Scan an integer
%lld	Scan a long long integer
%s	Scan a string
%c	Scan a character

Spaces and such

▶ Literal Characters

```
0 // will read "(10,20)"
1 scanf("(%d,%d)");
```

▶ Spaces

```
0 // will read "(10,20)", " ( 10, 20 )", but not "(10 ,20)"
1 scanf(" ( %d, %d )");
```

▶ A binary followed by vowels

```
0 // will read "110101 eieio"
1 scanf("%[01] %[aeiou]");
```

Interactive Tests

▶ Not common yet, but ICPC is starting to use them.

▶ One rule: call flush(stdout) every time you print.

```c
0 #include <stdio.h>
1
2 int main() {
3   int x,y;
4   while (scanf("%d %d",&x,&y) != EOF) {
5     if (x==-1 && y==-1)
6       break;
7     printf("%d\n",x+y);
8     fflush(stdout);
9   }
10 }
```