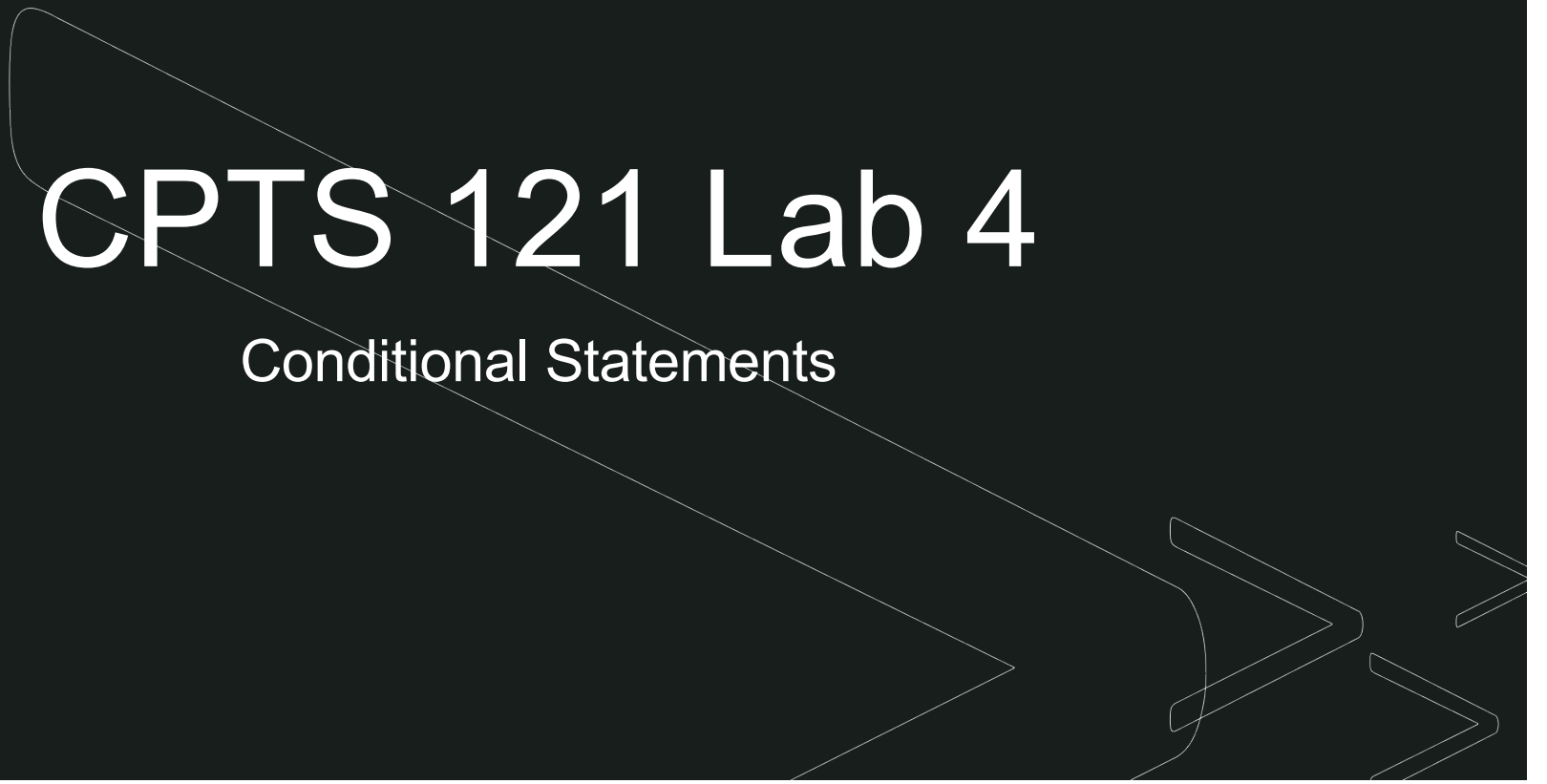
CPTS 121 Lab 4

Conditional Statements



if else

```c
int x = 5, y = 10;

    if (x < y)
        printf("x is less than y!");
    else
        printf("x is more than y!");
```

VALID

```c
int x = 5, y = 10;

if (x < y)
    printf("x is less than y!");
    printf("y is greater than x!");
else
    printf("x is more than y!");
```

INVALID

If Else Statements

- If statements are logical statements that control the flow of the code, hence they are a control statement
- If statements that only need to execute a single line do not need curly braces
- If statements do not need else or else if when there is no need
- Else statements are a catch-all statement
- All conditionals must be contained within parenthesis

Else If Statements

- Statements that will only check the next conditional if the first was false
- There can be many `else if` statements within the same branch

Else If

```c
int x = 5, y = 10;

    if (x < y) {
        printf("X is less than Y.\n");
    } else if (x == y) {
        printf("X equals Y.\n");
    } else {
        printf("Y is less than X.\n");
    }
```

```c
if (x < y) {
    printf("X is less than Y.\n");
} else if (x == y) {
    printf("X equals Y.\n");
} else if (x == 0 && y == 2) {
    printf("X is 0, y is 2.\n");
} else if (x == 1 && y == 4) {
    printf("X is 1, y is 4.\n");
} else if (x == -2 && y == 5) {
    printf("X is -2, y is 5.\n");
} else if (x == 213 && y == 52) {
    printf("X is 213, y is 52.\n");
} else if (x == -985 && y == 222) {
    printf("X is -985, y is 222.\n");
} else if (x == 52 && y == 24) {
    printf("X is 52, y is 24.\n");
} else if (x == 51 || y == 28) {
    printf("X is 51 OR y is 28.\n");
} else {
    printf("None of the conditions were met.");
}
```

Switch

```
int x = 2;

    switch (x) {
        case 1:
            someFunc();
            break;
        case 2:
            anotherFunc();
        case 3:
            finalFunc();
            break;
        default:
            break;
    }
```

```
typedef enum : int {
    FIRST_SOMETHING,  // 0
    SECOND_SOMETHING,  // 1
    THIRD_SOMETHING,  // 2

} MyEnum;


MyEnum e = FIRST_SOMETHING;


switch (e) {
    case FIRST_SOMETHING:
        someFunc();
        break;
    case SECOND_SOMETHING:
        anotherFunc();
    case THIRD_SOMETHING:
        finalFunc();
        break;
    default:
        break;
}
```

Switch Statements

- The variable must be an integral or enumerated type (simply put, a number)
- A single conditional statement
- Multiple case statements
  - A `break` statement must be used in every case unless you want to “fall through”
- `default` is the equivalent `else` clause
  - `default` should always be included to ensure a condition gets caught