CPTS 121 Lab 2

Methods

Top Down Design

- Start with the program's big problem
- Break it down into sub problems
- Break the sub problems into smaller parts known as algorithms
- Consider breaking down into smaller algorithms

Overview of (Single) C File With Methods

1. Include Statements
2. Define Statements
3. Method Stubs
4. Entry Method (int main(void) {})
5. Implementation of Methods

Overview (Code)

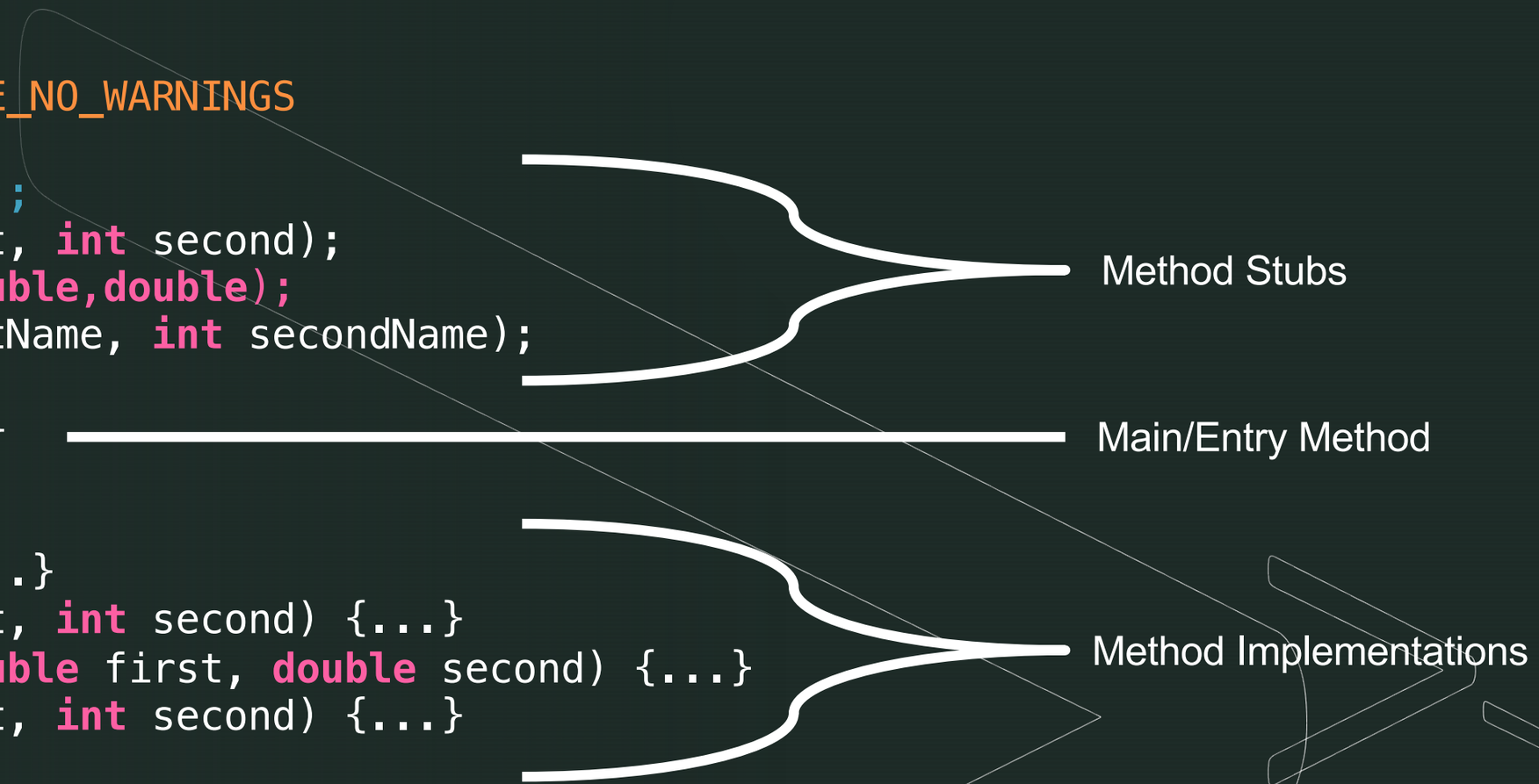
```c
#include <stdio.h>

#define _CRT_SECURE_NO_WARNINGS

void myMethod(void);
int myAdd(int first, int second);
double myDivide(double,double);
int mySub(int firstName, int secondName);

int main(void){...}


void myMethod() {...}
int myAdd(int first, int second) {...}
double myDivide(double first, double second) {...}
int mySub(int first, int second) {...}
```



Method Stubs (Code)

```
void myMethod(void);

    void myMethod();

int myAdd(int first, int second);

double myDivide(double,double);

int mySub(int firstName, int secondName);
```

Main(void)

OUTPUT:

```
myMethod Call!
myMethod Ran!


myAdd Call #5
myAdd Call #4
myAdd Call #3
myAdd Call #2
myAdd Call #1
ADD = 3


DIVIDE = 4.000000


SUB = 2

Program ended with exit code: 0
```

```c
int main(void) {
    // insert code here...
    printf("myMethod Call!\n");
    myMethod();

    printf("\n\n");

    int resultAdd = myAdd(1, 2);

    printf("ADD = %d \n\n", resultAdd);

    double resultDivide = myDivide(12, 3);

    printf("DIVIDE = %lf \n\n", resultDivide);

    int resultSub = mySub(3, 1);

    printf("SUB = %d\n", resultSub);

    return 0;
}
```

void myMethod(void)

OUTPUT:

```
myMethod Call!
myMethod Ran!
```

```c
void myMethod() {
    printf("myMethod Ran!\n");
    return;
}
```

int myAdd(int first, int second)

OUTPUT:

```
myAdd Call #5
myAdd Call #4
myAdd Call #3
myAdd Call #2
myAdd Call #1
ADD = 3
```

```c
int myAdd(int first, int second) {

    if (first < 5) {
        myAdd(first + 1, second);
    }

    printf("myAdd Call #%d\n", first);

    return first + second;
}
```

double myDivide(double,double)

OUTPUT:

DIVIDE = 4.000000

```
double myDivide(double first, double second) {
    double result = first / second;

    return result;
}
```

int mySub(int firstName, int secondName)

OUTPUT:

SUB = 2

```
int mySub(int first, int second) {
    return first - second;
}
```

Important Dates

- Tonight by Midnight
  - PA 1 Due
- Wednesday, Sept. 14 by Midnight [5 days]
  - PA 2 Due
- Friday, Sept. 16 by 5:10pm [7 days]
  - Quiz 3