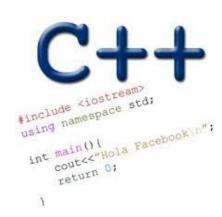# C++ PROGRAM DESIGN
TEST DRIVEN DEVELOPMENT

Problem Solving with Computers-I

https://ucsb-cs16-wi17.github.io/

ASCII Art: Write a program to draw a house of stars!

- Inputs: Dimensions of the house (width, height)
- Output: Drawing of a house made of ascii '*'

Step 1: Top-down design

- Break down the problem into subtasks

- Design your functions

```
string getBody(int width, int height);
string getRoof(int rheight);
string getHouse(string roof, string body);
```

Step 2: Implement and test each part

- Test code drives implementation- you are in control because you can….
  - measure your progress
  - write code systematically
  - debug systematically
  - automate testing

How to do TDD?

- Test suite: Bunch of tests
  - **Case 1: Is** `getRoof(0) == '*'`
  - **Case 2: Is** `getRoof(1) ==`
  - **Case 3: Is** `getRoof(2) ==`

`string getRoof(int rheight);`

- Test harness: Functions to report PASS/FAIL

  If expected == actual, report TEST PASSED!

  else report TEST FAILED!

What is returned by getRoof(1)

```
string getRoof(int height){
    int numSpaces;
    int numStars;
    string result;
    string spaces="";
    string stars="";

    for(int row = 0 ; row <=height;row++){
        numSpaces=0;
        numStars=1;
        spaces=stars="";

        for(int i=0;i<numSpaces;i++){
            spaces+=" ";
        }
        for(int i=0;i<numStars;i++){
            stars+="*";
        }

        result= result+spaces+stars+spaces+"\n";
    }
    return result;
}
```

A   *

B   *
       ***

C   *
    *

D   None of the above

Choose the replacement code to return the correct number of stars on each row for getRoof(1)

```
string getRoof(int height){
    int numSpaces;
    int numStars;
    string result;
    string spaces="";
    string stars="";

    for(int row = 0 ; row <=height;row++){
        numSpaces=0;
        numStars=1;
        spaces=stars="";

        for(int i=0;i<numSpaces;i++){
            spaces+=" ";
        }
        for(int i=0;i<numStars;i++){
            stars+="*";
        }

        result= result+spaces+stars+spaces+"\n";
    }
    return result;
}
```

```
*
***
```

A   numStars=row;

B   numStars=height-row;

C   numStars=2*row+1;

D   numStars=2*height+1;

Choose replacement code to return the correct output for getRoof(0) and getRoof(1)

```
string getRoof(int height){
    int numSpaces;
    int numStars;
    string result;
    string spaces="";
    string stars="";
    ;

    for(int row = 0 ; row <=height;row++){
        numSpaces=0;
        numStars=                 ;
        spaces=stars="";

        for(int i=0;i<numSpaces;i++){
            spaces+=" ";
        }  1;
        for(int i=0;i<numStars;i++){
            stars+="*";
        }

        result= result+spaces+stars+spaces+"\n";
    }
    return result;
}
```

```
      *            *
                  ***
```

A `numSpaces=row;`

B `numSpaces=height-row;`

C `numSpaces=2*row+1;`

D `numSpaces=2*height+1;`

Modify the code to make a hollow body

```
string getBody(int width, int height){
    string result;
    for(int row = 0;row <height;row++){
        for(int col =0; col<width;col++){
            result+='*';


        }
      result+="\n";
    }
    return result;

}
```

Next time

- Separate compilation with makefiles
- Pointers