

FUNCTIONS, MORE LOOPS

Problem Solving with Computers-I

<https://ucsb-cs16-sp17.github.io/>

C++

```
#include <iostream>
using namespace std;

int main(){
    cout<<"Hola Facebook\n";
    return 0;
}
```



Clickers out – frequency AB

Functions: Basic abstraction in programs

- Functions keep you DRY!
- Three steps when using functions
 1. DECLARE
 2. DEFINE
 3. CALL

ASCII art! Nested loops and functions

Write a FUNCTION that draws a square of a given width, and use it in a program with the following runtime behavior:

```
./drawSquare
```

```
Enter the width of the square
```

```
5
```

```
*****
```

```
*****
```

```
*****
```

```
*****
```

```
*****
```

Draw a triangle

Which line of the drawSquare code
(show on the right) would you modify
to draw a right angled triangle

```
./drawTriangle
Enter the length of the base
5

*
**
***
****
```

```
5 void drawSquare(int side){//A
6
7     for(int j = 0; j < side; j++){//B
8         for(int i=0; i < side; i++){//C
9             cout<<"*";
10        }
11        cout<<endl;
12    }
13    cout<<endl;
14
15 }
```

//D: A and B

//E: A and C

Passing parameters to programs

```
int main(int argc , char * argv[]){  
    ...  
}
```

```
$/drawTriangle
```

```
Enter the length of the base
```

```
5
```

```
*
```

```
**
```

```
***
```

```
****
```

```
$/drawTriangle 5
```

```
*
```

```
**
```

```
***
```

```
****
```

What is the value of argc in each of the following cases?

```
int main(int argc , char * argv[]){
```

```
...
```

```
}
```

```
$/drawTriangle
```

```
$/drawTriangle 5
```

```
$/drawTriangle 5 cat dog fizz
```

Control Flow: while and do while loops

```
while(Boolean expression) {  
    //statement 1  
    //statement 2  
}
```

```
do{  
    //statement 1  
    //statement 2  
}while(Boolean expression);
```

Evaluating C++ expressions with mixed types

```
int i =10;  
double sum = 1/i;
```

What is printed by the above code?

- A. 0
- B. 0.1
- C. 1
- D. None of the above

Setting up output when printing doubles

```
int i =10;  
double sum = 1/static_cast<double>(i);  
cout.setf(ios::fixed);      // Using a fixed point representation  
cout.setf(ios::showpoint); //Show the decimal point  
cout.precision(3);  
cout<<sum;
```

What is printed by the above code?

- A. 0
- B. 0.1
- C. 0.10
- D. 0.100
- E. None of the above

Write a FUNCTION that calculates the series:
 $1 + 1/2 + 1/3 + \dots + 1/n$, where `n` is a parameter passed to the program

Sample run of the program:

```
./sumseries 2  
Sum of the first 2 terms is : 1.500
```

```
./sumseries 3  
Sum of the first 3 terms is : 1.833
```

Identify the code that is not equivalent to the other two?
Assume 'n' is an integer that has already been declared (may be positive or negative)

A.

```
for( int x = 0; x < n; x++ ) {  
    cout<<x <<endl;  
}
```

B.

```
int x = 0;  
while(x < n) {  
    cout<< x << endl;  
    x++;  
}
```

C.

```
int x = 0;  
do{  
    cout<< x<< endl;  
    x++;  
} while(x < n);
```

D. They are ALL equivalent

Infinite loops

```
for(int y=0;y<10;y--)  
    cout<<"Print forever\n";
```

```
int y=0;  
for(;;y++)  
    cout<<"Print forever\n";
```

```
int y=0;  
for(;y<10;);  
    y++;
```

```
int y=0;  
while(y<10)  
    cout<<"Print forever\n";
```

```
int y=0;  
while(y=2)  
    y++;
```

Use while loops to print a sequence: x_{\min} , $x_{\min}+1$, $x_{\min}+2$,
.... x_{\max} for user specified inputs x_{\min} and x_{\max}

Sample run of the program: (You must use while loops, and specify the limits of the sequence as inputs to your program)

```
$ ./print_series 10 15  
10, 11, 12, 13, 14, 15
```

Function call mechanics

What is the output of the following code

```
void sum(int a, int b){  
    cout<< a+b;  
}  
  
int main(){  
    int result =0;  
    int x =10, y =20;  
    result = sum(x, y);  
    cout<<result+30;  
}
```

for loop OR while loop? Which one should you use?

```
for (int i = 0; i < 15; i++) {  
    cout << i << endl ;  
}
```

```
int j =0;  
while(j < 15) {  
    cout << j << endl ;  
    j++;  
}
```

Next time

- Automating the compilation process with Makefiles
- Intro to lab02