C/C++ MEMORY MODEL
FUNCTIONS
VARIABLE SCOPE AND LIFETIME

Problem Solving with Computers-I

Clickers out – frequency AB

Functions: Basic abstraction in programs

- Keep programs DRY !
- Three steps when using functions
  1. DECLARE: void drawSquare(int y);
  2. DEFINE: Write the actual code inside the function
  3. CALL: drawSquare(20);

  You must always declare/define functions before calling them.
  Demo the use of functions, print vs. return

Print vs return

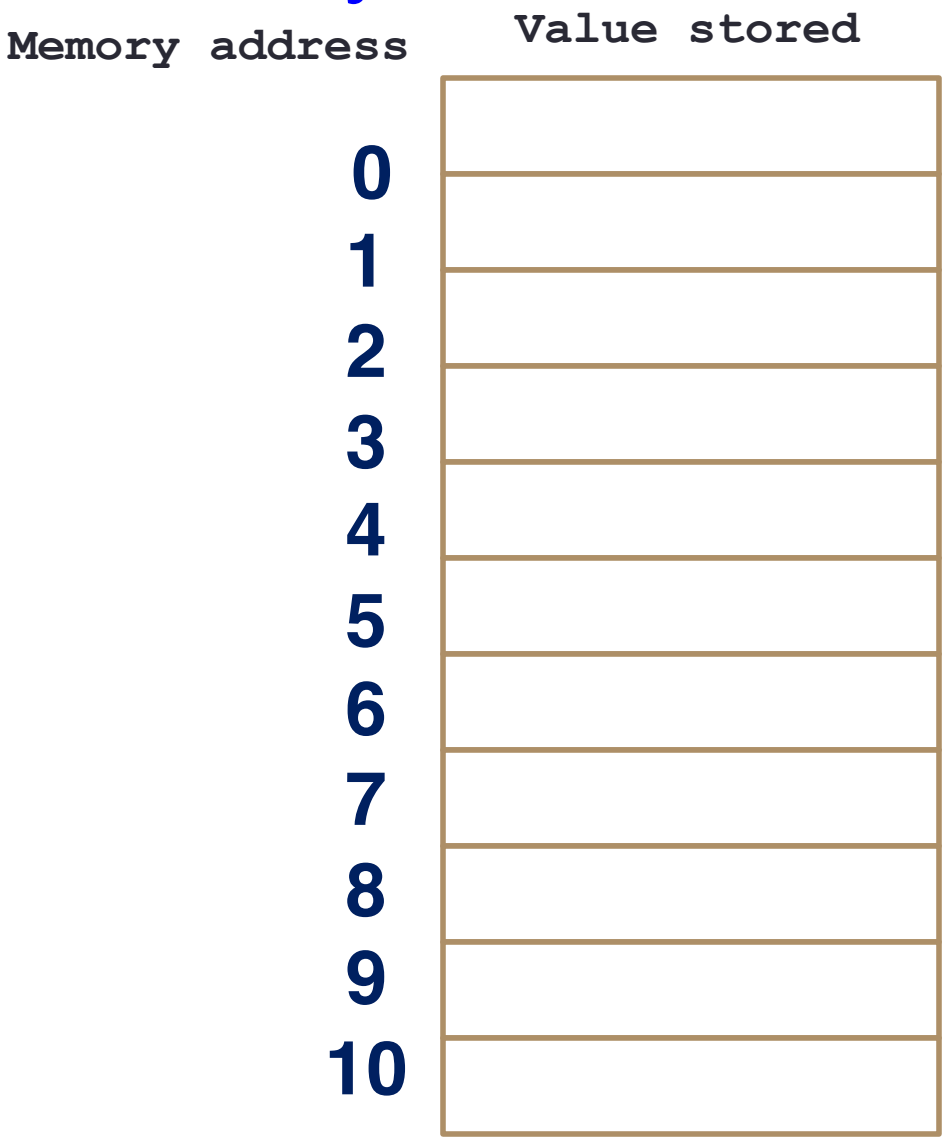
What is the output of the following code

```
int sum_1(int a, int b){
        return a+b;
}
int main(){
        int result =0;
        int x =10, y =20;
        result = sum_1(x, y); //What happens if we replace this
                              //sum_2?
        cout<<result;
        return 0;
}
```

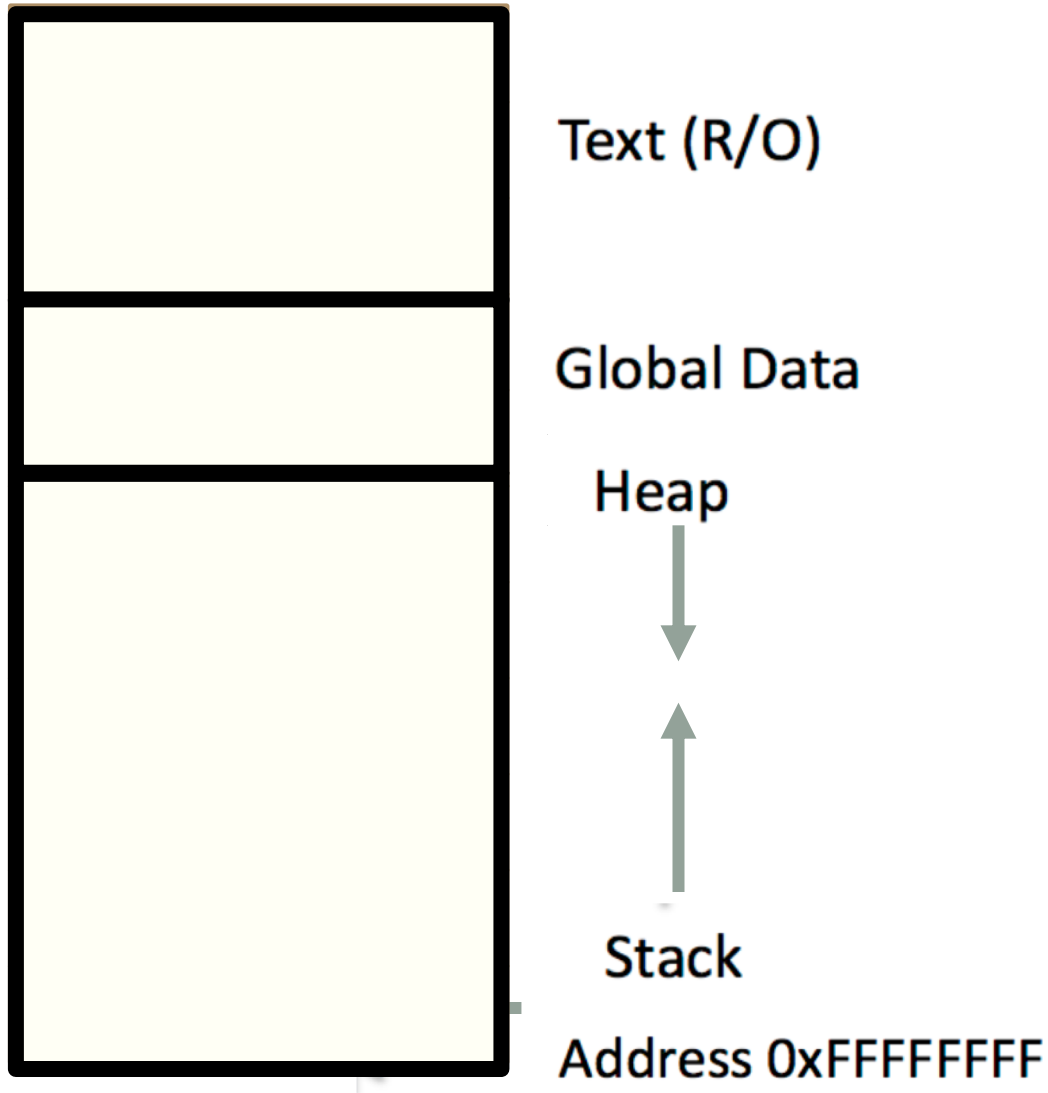
```
void sum_2(int a, int b){
        int result= a+b;
        cout<<result;
}
```

General model of memory

- Sequence of adjacent cells
- Each cell has 1-byte stored in it
- Each cell has an address (memory location)



C++ Memory Model



Stack: A region in program memory to "manage" local variables

Every time a function is called, its local variables are created on the stack

When the function returns, local variables are removed from the stack

Local variables are created and deleted on the stack using a Last in First Out principle

Variable: scope: Local vs global

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

int b_out;

int bar(){
      b_out = 20;
      int a_in = b_out+5;
      return a_in;

}
int main(){
   int result = bar();
   cout<<result;
   return 0;
}
```

Which variables are in memory when the cout<<result statement is executed?

A. result
B. b_out
C. a_in
D. A and B
E. None of the above

Pass by value

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

void bar(int x){
      x = x+5;

}

int main(){
   int y = 0
   bar(y);
   cout<<y;
   return 0;
}
```

What is printed by this code?

A. 0
B. 5
C. Something else

The C++ string class methods

- `string fruit = "Apple";`
- `int len = fruit.length();`
- `int pos= fruit.find('l');`
- `string part= fruit.substr(1,3);`
- `fruit.erase(2,3);`
  `fruit.insert(2,"ricot");`
- `fruit.replace(2,5,"ple");`
- `fruit = fruit+ " tasty";`

- Check out cctype for checks and conversions on characters
- `fruit[0]= tolower(fruit[0]);`
- `isalpha(fruit[0])`
- `isalnum(fruit[0])`

What is the output of the code?

```
string s1 = "Mark";
string s2 = "Jill";
for (int i = 0; i <= s1.length(); i++)
      s2[i] = s1[i];
if (s1 == s2) s1 = "Art";
cout<<s1<<" "<<s2<<endl;
```

A. Mark Jill

B. Mark Mark

C. Art Mark

D. Compiler error

E. Run-time error

Next time

- Automating the compilation process with Makefiles
- Intro to lab02