University of Southern California

Viterbi School of Engineering

Software Design

Function Pointers

Reference: Online Resources

Function Pointer

- In C/C++ a pointer does not need to necessarily point to data. We can have pointers that point to functions
- Similarly to variables, functions reside at an assigned address in memory
- A function pointer is a variable that points to the memory address where the function resides
- Remember that calling a function (i.e., the function name followed by the () operator) makes the execution jump to the memory location where the function resides
- Use of functions pointers improves the program efficiency and elegance
- When compared to a normal pointer, it is less prone to error, because memory will not be allocated/deallocated with them

Function Pointer – Example

- int (*myFuncPtr)();
  - myFuncPtr is a pointer to a function with no argument parameters and returns an integer value
  - myFuncPtr can point to any function that matches this function type

```
int myFunc1()  {  return 1;  }
int myFunc2() {  return 2;  }
int main()
{
   int (*myFuncPtr)() = myFunc1; // fcnPtr points to  myFunc1
   myFuncPtr = myFunc2 ; // fcnPtr now points to myFunc2
// Note: myFuncPtr = myFunc2() would be wrong, cause it puts the return value into myFuncPtr, which is wrong.
   return 0;
}
```

One Motivation for Function Pointers

- Question: What if we do not know yet at compile (build) time which function should be called? What if at run-time we decide to choose one function among the pool of functions to call?
  - Use switch statements (if/else, etc.)
    - Tedious for hundreds of functions

Example

```
bool (*compare) (int, int);
bool leq(int a, int b) { return a <= b; }
bool geq(int a, int b) { return a >= b; }
compare = geq; //Dereference pointer to function to execute
(*compare) (4,5);
```

Question: or is it (compare) (4,5); ?

Exercise

- Try fp and compare with switch-based

Arrays of Function Pointers

- An array of pointers to 10 functions:
  - int (*f[10])(void);
- This should work as well:
  - typedef int (*funPtr)(void);
  - funPtr F[10];

Exercise

- Implement an array of 3 function pointers
- Analyze qsort