
CS211 3-22-12

typedef:

Allow you to create alias for a data type, or a function.

Ex:

```
typedef double Currency;
typedef string FiveString[5];
typedef double Func(double);
```

Integration:

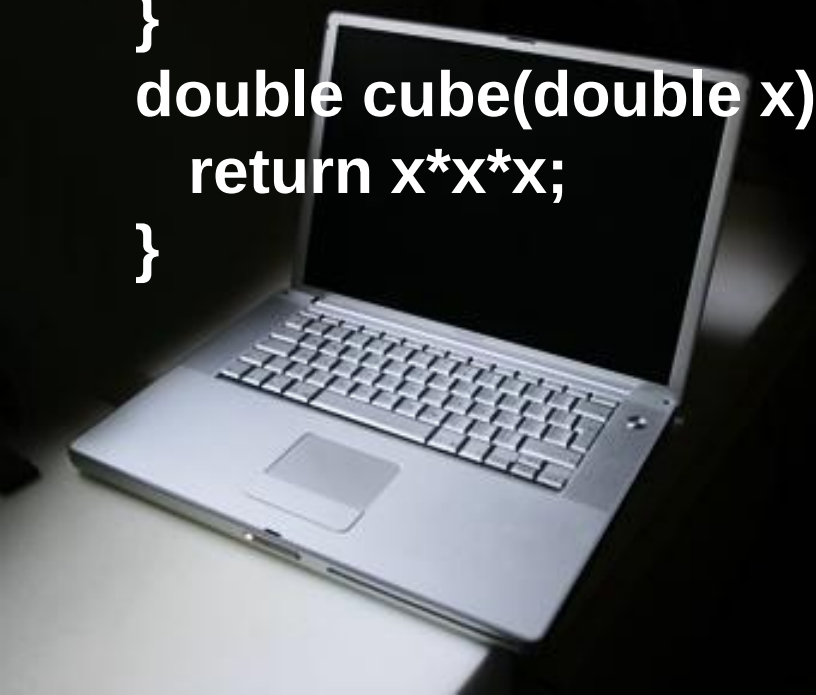
In math we can integrate a polynomial to figure out its area under the curve, giving the boundary values.

The idea behind integration is to create little tiny rectangles under the curve and sum up all those areas.

The tinier the rectangles are, the more precise the ‘integration’ result will be.

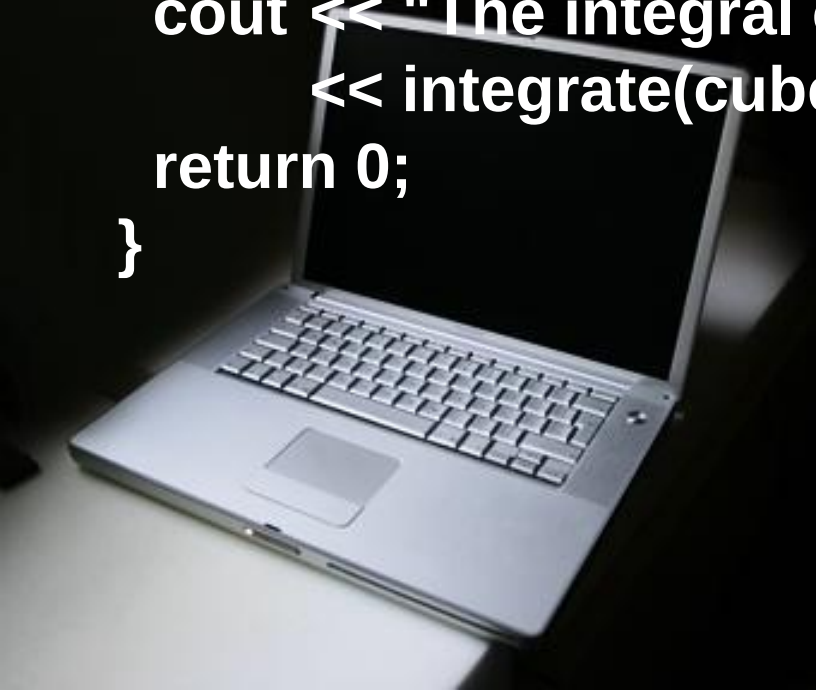
Integration Example:

```
double offset_value = .0001;
double line(double x){
   return x;
}
double square(double x){
   return x*x;
}
double cube(double x){
   return x*x*x;
}
```



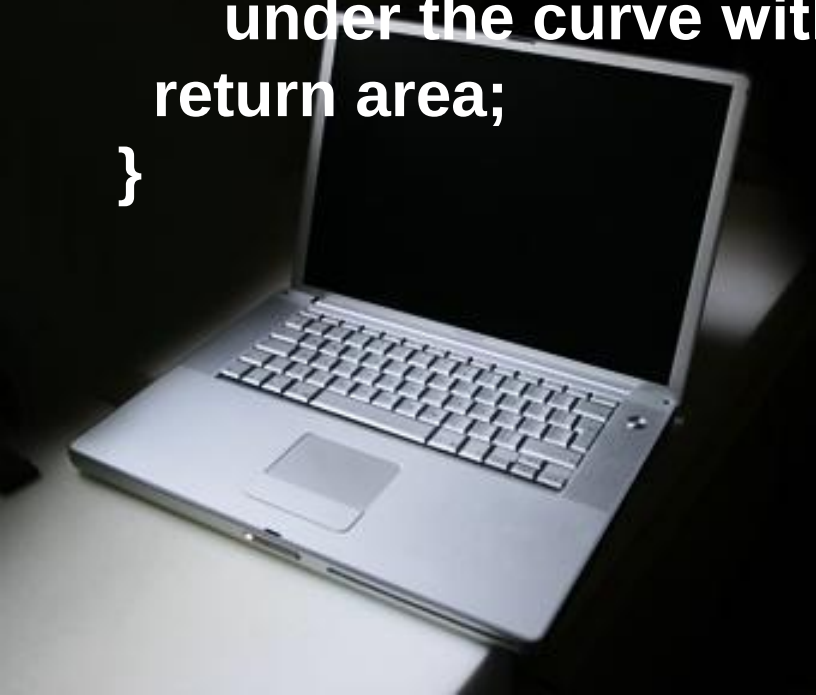
Integration Example:

```
int main()
{
  cout << "The integral of f(x) = x between 1 and 5 is: "
       << integrate(line, 1, 5) <<endl;
  cout << "The integral of f(x) = x^2 between 1 and 5 is: "
       << integrate(square, 1, 5) <<endl;
  cout << "The integral of f(x) = x^3 between 1 and 5 is: "
       << integrate(cube, 1, 5) <<endl;
  return 0;
}
```



Integration Example:

```
typedef double FUNC(double);
double integrate(FUNC f, double a, double b)
{
  double area = 0;
  // figure out the area under the curve
  /* loop from a to b, sum up all the computed little area
      under the curve with base as “offset_value” */
  return area;
}
```



Bubble Sort: - ascending order

```
for ( int j = 0 ; j < size - 1; ++j )
   for ( int i = 0 ; i < size - j - 1 ; ++i ) {
      if ( array[i] > array[i+1] ) {
         int temp = array[i];
         array[i] = array[i+1];
         array[i+1] = temp;
      }
   }
```

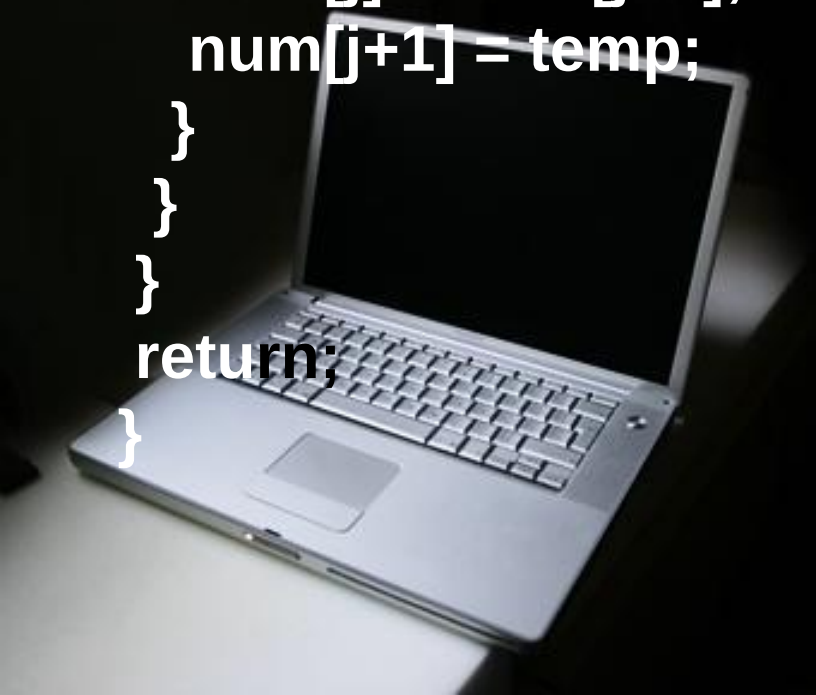
Function Pointer Example:

- Bubble Sort:
  Sort in ascending or descending order?

```
typedef bool ( *comparator ) ( int, int );
bool ascending(int k, int l)
{
 return k<l;
}
bool descending(int k, int l)
{
 return k>l;
}
```

Function Pointer Example:

```
void sort(int num[], int size, comparator comp){
 int temp;
 for(int i = 1; i <= size ; i++) {
  for (int j=0; j < (size -1); j++) {
   if ( comp(num[j+1],num[j]) ) {
    temp = num[j]; // swap elements
    num[j] = num[j+1];
    num[j+1] = temp;
   }
  }
 }
 return;
}
```



Function Pointer Example:

```
int main()
{
 int a[6] = {10, 100, 20, 90, 50, 25};
 comparator func = descending;
 sort(a, 6, func);
 for (int n = 0; n < 6; n++)
  cout << a[n] << " ";
 cout << endl;
 return 0;
}
```