

SSE2034: System Software Experiment 3 Spring 2016

Jinkyu Jeong (jinkyu@skku.edu)

Computer Systems Laboratory

Sungkyunkwan University

<http://csl.skku.edu>

Dynamic Memory

■ Dynamic memory

- Allocate *heap* memory at runtime
- Free the allocated memory

```
int *ptr = new int;           // allocate space for int
*ptr = 3;                     // set int value to 3

delete ptr;                   // free space for int
ptr = nullptr;                // purge out old pointer value
```

■ Local variables

- Allocated at *stack* on function entry
- deallocated from *stack* on function exit

[Lab – Practice #1]

■ Sorted-insert

- Use dynamic array
- Input:
 - The length of sequence
 - Numbers
- Output:
 - Sorted sequence of input numbers

```
$ ./sortedInsert
Length of array: 3
input: 5
5
input: 2
2 5
input: 3
2 3 5
```

[Lab – Practice #2]

▪ Implement car dealer management program

- Implement “car” class
- Each car has two properties: car id(unique), price
- Recommend to use array of pointers (100 elements)
- Input :
 - buy (car id) (price)
 - sell (car id) (price)
- Output :
 - information of remaining cars
 - total profit

[Lab – Practice #2]

```
$ ./manager
input: buy 3 100
3 100
-100
input: buy 3 120
car id 3 is already in the list!
input: sell 2 200
car id 2 is not in the list!
input: buy 2 80
3 100
2 80
-180
input: sell 3 120
2 80
-60
input: sell 2 110
50
```