Lab 2

Data Structures
C++ for C Coders

한동대학교 김영섭교수
idebtor@gmail.com

return by reference
in-house programming principle

Lab 2: Return by reference

- Modify the following programs such that it sets the maximum element to zero in main().

```
int max(int a[], int n) {
  int x = 0;
  for (int i = 0 ; i < n; i++)
    if (a[i] > a[x]) x = i;
  return a[x];
}

int main() {
  int a[] = {12, 42, 33, 99, 63};
  int n = 5;

  for (int i = 0; i < n; i++)
    cout << a[i] << " ";
}
```

```
12 42 33 0 63
```

Lab 2: Return by reference

- Modify the following programs such that it sets the maximum element to zero in main().

Do you remember the in-house programming principles?

```
int& max(int a[], int n) {
  int x = 0;
  for (int i = 0 ; i < n; i++)
    if (a[i] > a[x]) x = i;
  return a[x];
}

int main() {
  int a[] = {12, 42, 33, 99, 63};
  int n = 5;

  max(a, n) = 0;
  for (int i = 0; i < n; i++)
    cout << a[i] << " ";
}
```

12 42 33 0 63

Lab 2: Return by reference

- Modify the following programs such that it sets the maximum element to zero in main().

Do you remember the in-house programming principles?

Any violation of the programming principles in this code?

```
int& max(int a[], int n) {
  int x = 0;
  for (int i = 0 ; i < n; i++)
    if (a[i] > a[x]) x = i;
  return a[x];
}

int main() {
  int a[] = {12, 42, 33, 99, 63};
  int n = 5;

  max(a, n) = 0;
  for (int i = 0; i < n; i++)
    cout << a[i] << " ";
}
```

12 42 33 0 63

Return by reference

- By default in C++, when a function returns a value, it is copied into stack. The calling function reads this value from stack and copies it into its variables.
- An alternative to “return by value” is “return by reference”, in which the value returned is not copied into stack.
- One result of using “return by reference” is that the function which returns a parameter by reference **can be used on the left side** of an assignment statement.

Lab 2

Data Structures
C++ for C Coders

한동대학교 김영섭교수
idebtor@gmail.com

return by reference
in-house programming principle