아주대학교 프로그래밍언어

201821048 미디어학과 이서영

2019/12/01 함수, 배열

# 소스코드 및 주석

/\*

[실습#7] 참조에 의한 호출을 이용한 난수 생성

요구사항 : 다음 슬라이드의 코드를 이용해서 정수 배열을 받아서 요소들을

난수로 채우는 함수를 작성하고 테스트하라.

난수는 라이브러리 함수인 rand()을 사용하여 생성한다

\*/

#include <stdio.h>

#include <stdlib.h>

#include <time.h>

void array\_fill(int\* A, int size);

void array\_print(int\* A, int size);

int main(void)

{

srand(time(NULL));

int data[10];

array\_fill(data, 10);

array\_print(data, 10);

return 0;

}

void array\_fill(int\* A, int size)

{

for (int i = 0; i < size; i++) {

A[i] = rand() % 1000 + 1;

}

}

void array\_print(int\* A, int n)

{

int i;

printf("A[]={ ");

for (int i = 0; i < n; i++) {

printf("%4d,", A[i]);

}

printf(" }\n");

}

/\*

[실습#8] 배열 요소 출력 함수

요구사항 : 정수 배열의 요소들을 화면에 출력하는 함수를 작성하고 테스트하라.

출력 형식은 A[]={1,2,3,4,5} 와 같은 형식이 되도록 하라

\*/

#include <stdio.h>

void array\_print(int\* A, int n);

int main(void)

{

int list[10] = { 1, 2, 3, 4 };

array\_print(list, 10);

return 0;

}

void array\_print(int\* A, int n)

{

int i;

printf("A[]={ ");

for (int i = 0; i < n; i++) {

printf("%2d,", A[i]);

}

printf(" }\n");

}

/\*

[실습#9] 배열을 이용한 급여 계산 함수

요구사항 : 직원들의 기본급이 배열A[]에 저장되어 있다. 배열 B[]에는 직원들의

보너스가 저장되어 있다. 기본급과 보너스를 합하여 이번 달에 지급할

월급의 총액을 계산하고자 한다. A[]와 B[]를 더하여 배열C[]에

저장하는 함수를 작성하고 테스트하라.

//지급 : 기본급, 식대, 상여금

//공제 : 국민연금, 소득세, 고용보험

//지급총액, 공제총액

//Payment amount, basic salary, meal, bonus

//Deduction, National Pension, Income Tax, Employment Insurance

기본급 + 식대 + 상여금 = 지급

국민연금 + 소득세 + 고용보험 = 공제

지급총액 - 공제총액 = 총 급여

출력하는 함수와 각 금액을 계산하는 함수를 만들었다

메인에서 각 함수를 부르고 값을 저장한다.

\*/

#include <stdio.h>

#define array\_num 10

void print\_array(const char\* array\_name, int\* array) {

//배열출력

printf("%s", array\_name);

printf(" =");

printf("\n {");

for (int i = 0; i < array\_num; i++) {

printf("%5d", array[i]);

}

printf("}\n");

}

void array\_fill\_salary(int\* A)

//기본급

{

for (int i = 0; i < array\_num; i++) {

A[i] = (rand() % 50) \* 10 + 200;

}

}

void array\_fill\_bonus(int\* A)

//식대

{

for (int i = 0; i < array\_num; i++) {

A[i] = rand() % 10 + 50;

}

}

void array\_fill\_mealpaid(int\* A) {

//상여금

for (int i = 0; i < array\_num; i++) {

A[i] = 40;

}

}

void fill\_addresult(int fir[], int sec[], int thir[], int result[]) {

printf("\n\*\*\*\*\*\*\*\*\*\*\*\*addresult\n");

for (int i = 0; i < array\_num; i++) {

result[i] = fir[i] + sec[i] + thir[i];

}

}

void fill\_national\_pension(int\* A) {

////공제 : 국민연금

for (int i = 0; i < array\_num; i++) {

A[i] = 60;

}

}

void fill\_income\_tax(int\* A, int\* B) {

//소득세

for (int i = 0; i < array\_num; i++) {

(int)B[i] = A[i] \* 0.1;

}

}

void fill\_employment\_insurence(int\* A) {

//고용보험

for (int i = 0; i < array\_num; i++) {

A[i] = 20;

}

}

void fill\_total\_salary(int fir[], int sec[], int result[]) {

//지급총액 - 공제총액 = 총 급여

printf("\*\*\*\*\*\*\*\*\*\*\*\*total\_salary\n");

for (int i = 0; i < array\_num; i++) {

result[i] = fir[i] - sec[i];

}

}

int main() {

int salary[array\_num];

array\_fill\_salary(salary);

print\_array("salary", salary);

int bonus[array\_num];

array\_fill\_bonus(bonus);

print\_array("bonus", bonus);

int mealpaid[array\_num];

array\_fill\_mealpaid(mealpaid);

print\_array("mealpaid", mealpaid);

int Payment\_amount[array\_num];

fill\_addresult(salary, bonus, mealpaid, Payment\_amount);

print\_array("Payment\_amount", Payment\_amount);

printf("\*\*\*\*\*\*\*\*\*\*\*\*addresult\n\n");

int national\_pension[array\_num];

fill\_national\_pension(national\_pension);

print\_array("national\_pension", national\_pension);

int income\_tax[array\_num];

fill\_income\_tax(salary,income\_tax);

print\_array("income\_tax", income\_tax);

int employment\_insurance[array\_num];

fill\_employment\_insurence(employment\_insurance);

print\_array("employment\_insurance", employment\_insurance);

int deduction\_amount[array\_num];

fill\_addresult(national\_pension, income\_tax, employment\_insurance, deduction\_amount);

print\_array("deduction", deduction\_amount);

printf("\*\*\*\*\*\*\*\*\*\*\*\*addresult\n\n");

int total\_salary[array\_num];

fill\_total\_salary(Payment\_amount, deduction\_amount, total\_salary);

print\_array("total\_salary", total\_salary);

printf("\*\*\*\*\*\*\*\*\*\*\*\*total\_salary\n");

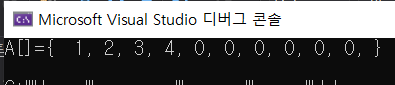
return 0;

}

No 7



No 8



No 9

