

컴퓨터 프로그래밍2

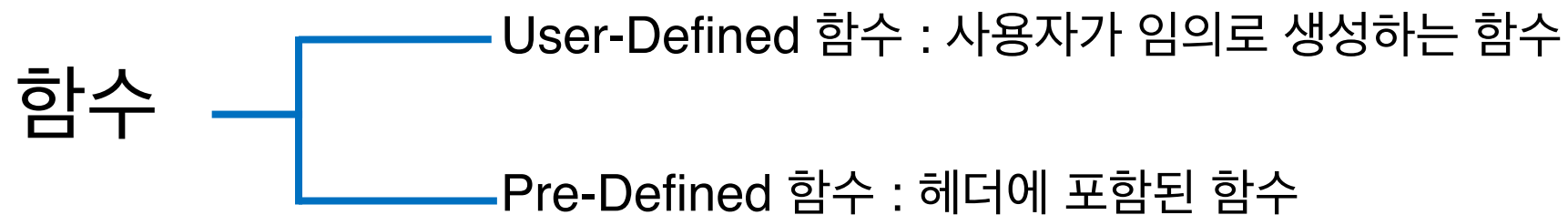
5주차_실습

충남대학교 컴퓨터공학과
장서윤

pineai@cnu.ac.kr

함수 이용의 장점

- ▶ 하나의 프로그램을 여러 개의 함수로 만드는 것은 효과적
- ▶ 특정 기능별로 분리하여 읽기 쉽고, 이해하기 쉽게 설계
- ▶ 한번 정의된 함수를 호출하여 사용되는 것이기에 프로그램의 양을 줄일 수 있음



수학 관련 라이브러리 함수

```
#include <stdio.h>
```

```
#include <math.h>
```

 — 헤더파일 : math 함수 포함 라이브러리

```
int main(void) {
```

```
    int n;
```

```
    double x, y;
```

```
    printf("정수를 입력하시오: ");
```

```
    scanf("%d", &n);
```

```
    printf("%d의 절대값 : %d\n", n, abs(n)); // int abs(int n)
```

 — 변수를 절대값으로
변경하는 함수(|x|)

```
    printf("실수 x를 입력하시오: ");
```

```
    scanf("%lf ", &x);
```

```
    printf("x³ : %lf\n", pow(x,3.0)); // double pow(double x, double y)
```

 — 지수 함수 (x^3)

```
    printf("64의 제곱근 : %lf\n", sqrt(64.0)); // double sqrt(double x)
```

 — 제곱근 함수

```
    return 0;
```

```
}
```

함수 프로그램의 구조

.....

```
long sum(int n)
```

함수 헤더로 이 함수의 이름은
sum()이다

```
{
```

```
    long result = 0;
```

```
    int i = 1;
```

```
    for(i = 1; i <= n; i++) {
```

```
        result += i;
```

```
    }
```

```
    return result;
```

함수 몸체로 해당 함수의 필요
한 작업을 구현한다.

```
}
```

함수 프로그램의 구조

```
int main(void) {  
    int n = 1, i = 1;  
    long result = 0;  
  
    //표준입력으로 정수 읽기  
    //A. 사용자 입력 정수 n을 표준입력으로 하나의 변수에 저장  
    printf("양의 정수를 하나 입력하세요. ");  
    scanf("%d", &n);  
  
    //짝수, 홀수 구분  
    //B. 위 변수의 값을 조사하여 짝수이면 합을 홀수이면 곱을 구하자.  
    if (n%2 == 0)  
    {  
        result = sum(n);           //함수 sum() 호출  
    } else  
    {  
        result = mult(n);          //함수 mult() 호출  
    }  
  
    //출력  
    //C. 결과를 출력  
    printf("%2d부터 %3d까지의 %s은 %3d입니다. \n", 1, n, (n%2) ? "곱" : "합", result);  
  
    return 0;  
}
```

재귀 함수

▶ 함수 내에서 자기 자신을 호출하는 함수

팩토리얼 함수

```
#include<stdio.h>

long factorial(int n); 함수 선언

int main(void)
{
    int i = 0;

    printf("n을 입력하세요!");

    scanf("%d", &i);

    printf("%2d! = %ld\n", i, factorial(i)); 함수 호출

    return 0;
}

long factorial(int n) 함수 정의
{
    if(n<=1)
        return 1;
    else
        return (n*factorial(n-1));
}
```

PRE-DEFINED 함수 프로그램

▶ 레지스터 변수 속도 확인 프로그램

```
#include <stdio.h>
#include <time.h>
#define MAX 1000000

int main(void)
{
    register int i;
    clock_t startTime, endTime, result;

    startTime=clock( );
    for (i=0; i<=MAX; i++)
    {
        printf("%d\\n", i);
    }
    endTime=clock( );

    result=endTime-startTime;
    printf("레지스터 변수 속도 : %lf초\\n", (double)result/1000);
    return 0;
}
```

연산속도 측정 함수 clock() 포함 헤더 – CPU 클럭 값 반환

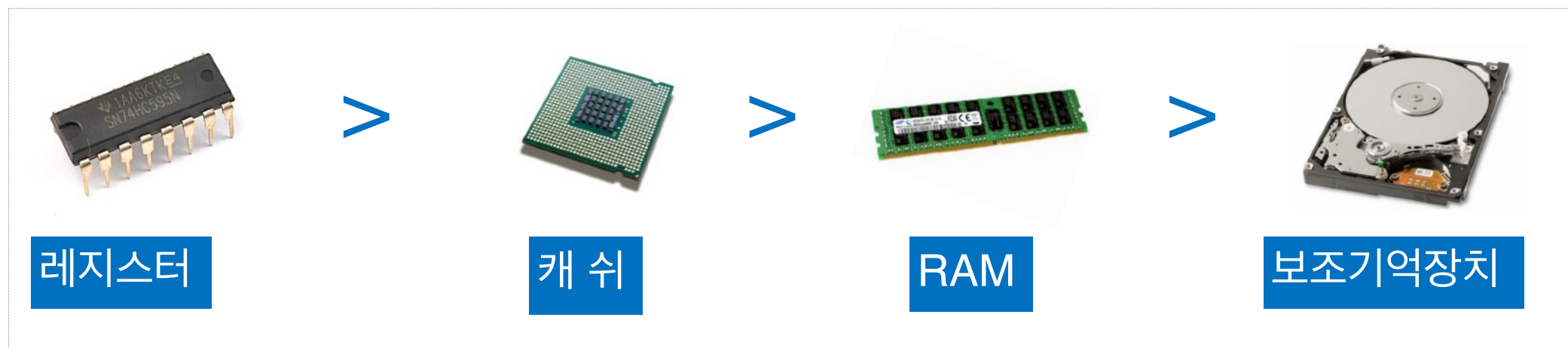
레지스터 타입 선언 // int i;

Clock 타입 변수 선언

연산속도 측정

레지스터 & RAM

- 레지스터
 - CPU 내부에 존재하는 기억장치
 - ‘저장’ 보다 ‘연산’ 을 위한 공간
- RAM
 - 휘발성 내용 기억장치
 - 실행되어질 프로그램과 데이터 저장 공간



CPU 안에 포함되어 있다보니 연산과정이나 결과를 레지스터에 입력하는 것이 더 빠름

PRE-DEFINED 함수 프로그램

.....

Register int : CPU 레지스터 이용

```
C:\Windows\system32\cmd.exe
999978
999979
999980
999981
999982
999983
999984
999985
999986
999987
999988
999989
999990
999991
999992
999993
999994
999995
999996
999997
999998
999999
1000000
레지스터 변수 속도 : 35.374000초
계속하려면 아무 키나 누르십시오 . . .
```

Int : 메모리(RAM) 이용 연산

```
C:\Windows\system32\cmd.exe
999978
999979
999980
999981
999982
999983
999984
999985
999986
999987
999988
999989
999990
999991
999992
999993
999994
999995
999996
999997
999998
999999
1000000
레지스터 변수 속도 : 39.676000초
계속하려면 아무 키나 누르십시오 . . .
```

실습 1

```
1 #include <stdio.h>
2
3 int aaa(int su1, int su2){ //매개변수로 두개의 int값 받음
4
5     int hap;           //반환할 값 int형 hap
6     hap=su1+su2;       // 두 수의 합
7     return hap;        //호출한 곳으로 반환할 int형 return값
8 }
9
10
11
12 int main (void){
13     int result;         //return 되는 값을 저장할 int형 변수
14     result=aaa(5,3);    //aaa() 함수 호출하면서 두개의 인자 전달, 결과값은 result에 저장
15
16     printf("aaa()함수를 호출하면서 인자로 5와 3 전달해서 결과는 %d이 됩니다.\n", result);
17     //반환되어 돌아온 int형 결과값 출력
18     return 0;
19 }
20
```

int 형 전달인자

int 형 리턴 값

실습 2

```
#include <stdio.h>
```

```
// 함수 선언
```

```
int add(int a, int b);
```

```
int subtract(int a, int b);
```

```
int main(void)
```

```
{
```

```
    int v1, v2, m=10, n=20;
```

```
    // 함수 호출
```

```
    v1 = add(m,n);
```

```
    v2 = subtract(15,3) + 2;
```

```
    printf("%d %d %d \n",
```

```
           v1, v2, add(2,6) );
```

```
    return 0;
```

```
}
```

```
// 함수 add 정의
```

```
int add(int a, int b)
```

```
{
```

```
    return (a+b);
```

```
}
```

```
// 함수 subtract 정의
```

```
int subtract(int a, int b)
```

```
{
```

```
    int diff;
```

```
    diff = a-b;
```

```
    return diff;
```

```
}
```

실습 3

▶ 사각형 넓이 출력 프로그램 구현

```
#include <stdio.h>
int getArea(int x, int y);

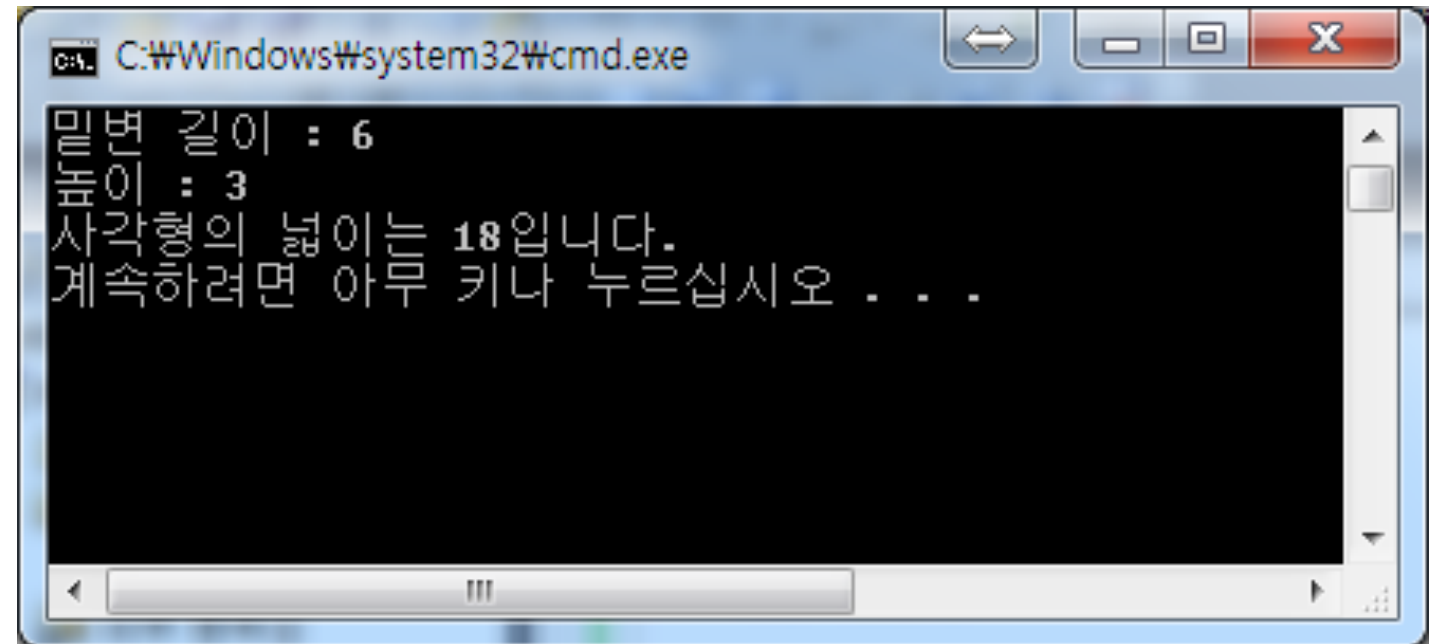
int main(void) {
    int a, b;
    int result;

    printf("밑변 길이 : ");
    scanf("%d", &a);
    printf("높이 : ");
    scanf("%d", &b);

    result = getArea(a, b);
    printf("사각형의 넓이는 %d입니다. \n", result);

    return 0;
}

int getArea(int x, int y) {
    return x * y;
}
```



실습 4

- ▶ 1부터 n까지의 합 구하는 프로그램 작성
 - ▶ 팩토리얼 재귀함수와 유사

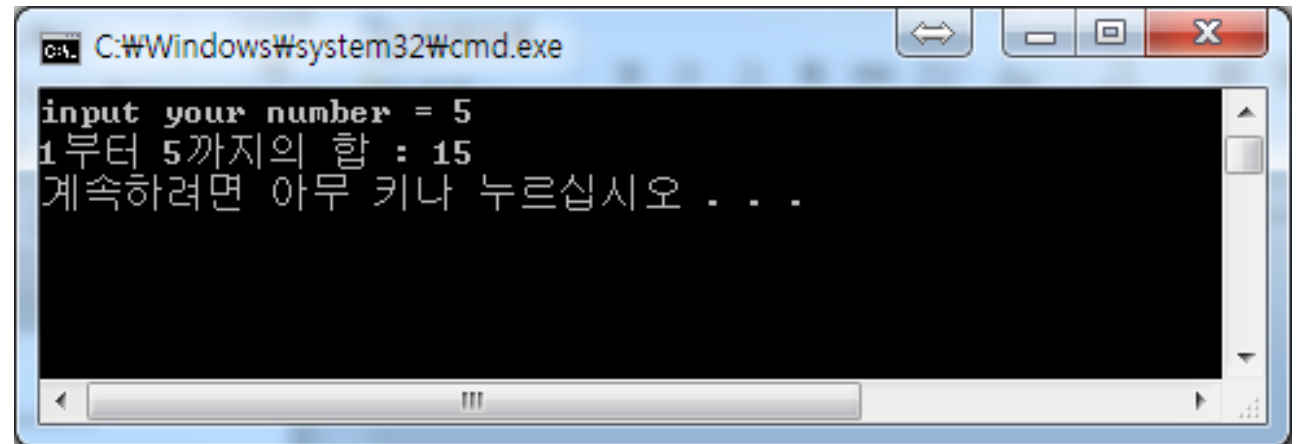
```
#include <stdio.h>
int sum(int x);

int main(void) {
    int a;
    int result;

    printf("input your number = ");
    scanf("%d", &a);
    result = sum(a);
    printf("1부터 %d까지의 합 : %d \n", a, result);

    return 0;
}

int sum(int x) {
    if(x <= 1)
        return 1;
    else
        return x + sum(x-1);
}
```



과제 1

- ▶ 2부터 입력받은 수까지의 모든 소수를 구하여 한 줄에 3개씩 출력하는 프로그램을 작성
- ▶ `int prime_check(int checkNum)`
 - ▶ 양수를 매개변수로 받아 소수인지를 판단하는 메소드

과제 2

▶ 다음의 시나리오를 완성하는 코드를 구현하시오.

▶ 재귀로 구현 되어야 함

토끼를 한 쌍 키우고 있다.

한 달이 지나면 새끼 토끼를 낳고, 새로 태어난 토끼는 한 달이 지나면 새끼를 낳는다.

그 토끼는 한 달이 지나면 어른 토끼가 되고, 어른 토끼는 새끼를 낳는다.

토끼가 태어난 지 1년 만에 죽지 않는다고 가정할 때,

토끼가 매달 새끼를 낳으면 1년 후에 전체 토끼 수가 얼마나 증가하게 되는지 알 수 있다.

▶ 실행 결과

1 1 2 3 4 8 13 21 34 55 89 144

추가 과제

- ▶ 음수가 아닌 숫자를 입력받아 해당 숫자의 피보나치수를 계산하는 프로그램
- ▶ 피보나치 수를 재귀로 구현

$$\text{fib}(0) = 1$$

$$\text{fib}(1) = 1$$

$$\text{fib}(n) = \text{fib}(n-1) + \text{fib}(n-2)$$

- ▶ 입력을 40이상으로 줄 경우 생기는 문제점을 해결하여 구현할 것
- ▶ 입력이 50이상일 경우 생기는 문제에 대해서 주석으로 작성하여 제출 할 것

과제제출

- ▶ 사이버캠퍼스에 제출
- ▶ 파일 이름 : [CP2]05_본인학번
- ▶ 제출 기한 : 다음주 목요일(12일) 23:59까지
- ▶ 소스에는 주석 작성이 되어 있어야 함
- ▶ 실습은 수행 결과를 [스크린샷]하여 결과에 대한 설명을 적어 제출 (제출 방법은 상관 없음)
- ▶ 과제는 소스와 스크린샷을 제출
- ▶ 실습과 과제 모두 압축하여 하나의 파일로 제출
- ▶ 추가과제는 메일로 직접 보낼 것 (메일 제목은 파일 이름과 동일)