

안녕하세요 선생님

현재 집에 pc가 없어 급히 친구 놈 노트북을 빌려서 하는 바람에

직접 우분투 리눅스를 설치 할 수 없어, 간단하게 명령어 정리 ,

오늘 배웠던 알고리즘 4개를 복습했습니다.

별거는 아니지만 이것이라도 해야 오늘 미진한 학습량을 맞출 수 있을 같아 열심히 하였습니
다 .. (개판입니다)

열심히 하겠습니다.

감사합니다

리눅스 기본 명령어

ls //리스트 보기

pwd //현재 사용자 위치

cd//이동 cd../상위 디렉토리로 이동 cd~//어디에있던 홈으로 이동

cp//복사 cp test1.c test2.c//test1.c를 test2라는 이름으로 복사

rm//삭제 rm-f//문지도 바로 삭제 rm -rf //삭제하는 이름

mkdir//디렉토리 생성

./ // 실행

gcc //컴파일(.c로 끝나야함) gcc -o 바꿀이름 본래파일이름

gdb //디버깅

cat//빠른실행

ctrl+alt+t//터미널 실행

Vi 명령어

dd// 1행씩 삭제

yy// 1행씩 복사

:set nu//행 번호 표시

%s/abc/aaa/g // abc를 aaa로 죄다 바꿈

git clone 주소 // 깃 허브에 업로드된 자료를 받는다.

:wq// 저장하고 나감

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

func1 알고리즘

```
int myfunc(int num)
```

```
{
```

```
    // return num * 2;
```

```
    return num + 3;
```

```
    // return num << 1;
```

```
}
```

```
int main(void)
```

```
{
```

```
    int num = 3, res;        //1.num에 정수3이라는 변수 선언 , res 정수 변수 선언
```

```
    res = myfunc(num);       //2.왼쪽 myfunc에 num 값을 대입해서 num +3을 반환
```

```
    printf("res = %d\n", res); //3. 6이라는 값이 나옴
```

```
    return 0;
```

```
}
```

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

func2 알고리즘

```
// input = 3, 6, output = 21.3
```

```
float myfunc(int num1, int num2)
```

```
{
```

```
    return num1 * num2 + 3.3;
```

```
}
```

```
int main(void)
```

```
{
```

```
    int num1 = 3, num2 = 6; //1.num1에 정수3이라는 변수선언 2에는 6
```

```
    float res;                // 실수형 res 변수 선언
```

```
    res = myfunc(num1, num2); //왼쪽 myfunc 에 num1,num2대입
```

```
    printf("res = %f\n", res);    // num1과num2를 곱하고 3.3을 더한값을 출력  
                                21.9999
```

```
    printf("this = %f\n", myfunc(num1, num2)); //num1과num2를 곱하고 3.3을 더  
                                                한값을 출력  
                                                21.9999
```

```
    return 0;
```

```
}
```

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

```
// input = 3, 6, output = 21.3
```

```
float myfunc(int num1, int num2)
```

func3 알고리즘

```
{
```

```
    return num1 * num2 + 3.3;
```

```
}
```

```
float test(int a, int b)
```

```
{
```

```
    printf("I can call myfunc()\n");
```

```
    return myfunc(a, b);
```

```
}
```

```
int main(void)
```

```
{
```

```
    int num1 = 3, num2 = 6; // 변수 선언
```

```
    printf("res = %.1f\n", test(num1, num2)); // test로 올라가서 대입
```

```
                                I can call myfunc
```

```
                                위쪽 myfunc 올라가서 대입
```

```
                                res=21.3
```

```
    return 0;
```

```
}
```

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

func4 알고리즘

```
int test(int num1, int num2, int num3, int num4, int num5)
```

```
{
```

```
    return num1 + num2 + num3 + num4 + num5;
```

```
}
```

```
int main(void)
```

```
{
```

```
    int num1 = 2, num2 = 4, num3 = 6, num4 = 8, num5 = 10, res;
```

```
    //num1부터 정수 변수 선언 2,4,6,8,10 res 선언
```

```
    res = test(num1, num2, num3, num4, num5);
```

```
    //위쪽 테스트에 대입 , 리턴값으로 더한값을 반환
```

```
    printf("res = %d\n", res);
```

```
    //모두 더한 값 res=30;
```

```
    return 0;
```

```
}
```