

## 1. 예제

### I . 예제\_1.1

- 주석 사용 연습 예제

```
/*  
    여기는 주석부분 입니다.  
    컴파일 시 이 부분은 컴파일 되지 않는  
    코드영역입니다.  
*/  
  
#include <stdio.h>  
  
int main(void)  
{  
    // 이것 또한 주석 입니다.  
    printf("Hello World!\n");  
  
    return 0;  
}
```

### II . 예제\_1.2

- 특수 문자 사용 연습 예제

```
#include <stdio.h>  
  
int main(void)  
{  
    printf("\t###          성적표          ###\n");  
    printf("=====\n");  
    printf("이름\t국어\t영어\t수학\t체육\t음악\n");  
    printf("-----\n");  
    printf("김연아\t95\t100\t76\t100\t97\n");  
    printf("박찬호\t60\t100\t75\t100\t77\n");  
    printf("유이\t90\t70\t70\t99\t90\n");  
    printf("안철수\t100\t100\t100\t66\t65\n");  
    printf("-----\n");  
}
```

```

        return 0;
    }

```

## 2. 문제

### I. 문제\_1.1

- 특수 문자 연습 문제

```

                *
            *
        *
    *
*

```

계속하려면 아무 키나 누르십시오 . . .

### II. 문제\_1.2

- 특수 문자 연습 문제

나보기가 역겨워 가실 때에는 "사뿐히" 즈려뵈고 가시옵소서.  
 나보기가 역겨워 가실 때에는 ~~사~~~~뿐~~~~히~~ 즈려뵈고 가시옵소서.  
 계속하려면 아무 키나 누르십시오 . . . ■

### III. 문제\_1.3

- 특수 문자 연습 문제

##### 세금 고지서 #####			
세 목	납기액	가산세	합계금액
주민세	165230	0	165230
지방세	65000	0	65000
부가세	5230	0	5230
총합계	235460		

계속하려면 아무 키나 누르십시오 . . . ■

## 1. 예제

### I . 예제\_2.1

- 정수, 실수, 문자형 변수에 각각 값을 대입한 후, 출력하는 예제

```
#include <stdio.h>

int main(void)
{
    int val1 = 100;
    double val2 = 200.0;
    char val3 = 'A'

    printf("val1은정수이며값은%d 입니다.\n",val1);
    printf("val2은실수이며값은%f 입니다.\n",val2);
    printf("val3은문자이며값은%c 입니다.\n",val3);

    return 0;
}
```

### II . 예제\_2.2

- 원의 넓이를 계산하여 출력하는 예제

```
#include <stdio.h>

int main(void)
{
    double pi = 3.141592;           // PI(파이)
    int radius = 7;                 // 반지름
    double result = pi*radius*radius; // 결과: 원의넓이

    printf("PI는%f입니다.\n", pi);
    printf("원의반지름은%d 입니다.\n", radius);
    printf("원의넓이는%.3f 입니다.\n", result);

    return 0;
}
```

### III. 예제\_2.3

#### - ASCII 코드 문자 출력 예제

```
#include <stdio.h>

int main(void)
{
    char lower = 'a';
    char upper = 'A';
    char number = '1';

    printf("입력문자: %c\n", lower);
    printf("ASCII 코드10진수: %d\n\n", lower);
    printf("ASCII 코드16진수: %x\n\n", lower);

    printf("입력문자: %c\n", upper);
    printf("ASCII 코드10진수: %d\n\n", upper);
    printf("ASCII 코드16진수: %x\n\n", upper);

    printf("입력문자: %c\n", number);
    printf("ASCII 코드10진수: %d\n\n", number);
    printf("ASCII 코드16진수: %x\n\n", number);

    return 0;
}
```

## 2. 문제

### I. 문제\_2.1

- 점수(val), 등급(grade)이란 변수를 선언한 후, 점수는 100으로 등급은 A로 초기화하여 화면에 아래와 같이 출력하는 프로그램을 작성

```
점수 : 100, 등급 : A
계속하려면 아무 키나 누르십시오 . . .
```

### I. 문제\_2.2

- 반지름의 길이가 2인 원의 넓이는 구하는 프로그램 작성

```
PI는3.141592입니다.
원의반지름은2 입니다.
원의넓이는12.566 입니다.
계속하려면 아무 키나 누르십시오 . . .
```

## 1. 예제

### 1. 예제\_3.1

- 출력 변환문자와 옵션 지정자를 이용하여 출력한 예제

```
#include <stdio.h>

int main(void)
{
    int i = 28;
    int j = 77;
    double d = 135.7979;
    char c = 'A'

    printf("%d %#o %#X\n", j, j, j);
    // 10진수를10, 8, 16진수로 출력(접두어표시: #);
    printf("%d %d %d \n", 10, 010, 0x10);
    // 10, 8, 16진수를10진수로 출력

    printf("%5d\n", i);
    // 5자리 우측 정렬
    printf("%5c\n", c);
    // 5자리 공간 확보 우측 정렬
    printf("%-5d%-5d\n", i, i);
    // 5자리좌측정렬
    printf("%08d\n", i);
    // 8자리 공간 확보- 빈공간은 0으로 채움
    printf("%8.2f\n", d);
    // 8자리 공간 확보- 소수점이하 2자리 출력
    printf("100% success\n");
    // 문자'%' 출력

    return 0;
}
```

---

## II . 예제\_3.2

- 입력 변환 문자를 이용하여 정수와 실수 두 개씩을 입력받아 평균을 구하는 프로그램 예제

```
#include <stdio.h>

int main(void)
{
    int d1, d2;
    double f1, f2;
    double re1, re2;
    printf("정수 두 개 입력: ");
    scanf("%d %d", &d1, &d2);
    printf("실수 두 개 입력: ");
    scanf("%lf %lf", &f1, &f2);           // double형실수입력

    re1 = (d1 + d2) / 2.0;                 // 정수 연산의 결과는 정수
    re2 = (f1 + f2) / 2;
    printf("\n정수 평균%.2f\n", re1);       // 소수점이하2자리출력
    printf("실수 평균%.2f\n", re2);

    return 0;
}
```

## 2. 문제

### I . 문제\_3.1

- 국어(kor), 영어(eng), 수학(math) 점수를 입력 받아 합계(sum)를 계산하여 화면에 출력하는 프로그램 작성

```
[국어] [영어] [수학] 점수 입력 :89 76 98

[국어] : 89
[영어] : 76
[수학] : 98
[합계] : 263
계속하려면 아무 키나 누르십시오 . . .
```

## 1. 예제

1. 예제\_4.1

- 조건 연산자를 이용한 예제

```
#include <stdio.h>

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

    printf("첫번째수를입력하세요");
    scanf("%d", &a);
    printf("두번째수를입력하세요");
    scanf("%d", &b);

    large = a>b ? a : b;

    printf("%d와%d중큰수는%d입니다\n", a, b, large);

    return 0;
}
```

---

## II . 예제\_4.2

- sizeof연산자를 이용한 예제

```
#include <stdio.h>

int main(void)
{
    double a = 3.14;
    short b = 3;

    printf("int = %dbyte\n", sizeof(int) );
    printf("short = %dbyte\n", sizeof(short) );

    printf("변수a = %dbyte\n", sizeof(a) );
    printf("변수b = %dbyte\n", sizeof(b) );

    printf("정수상수5 = %dbyte\n", sizeof(5) );
    printf("5+4 = %dbyte\n", sizeof(5+4) );

    printf("실수상수123.12 = %dbyte\n", sizeof(123.12) );
    printf("실수상수123.12f = %dbyte\n", sizeof(123.12f) );

    return 0;
}
```

## II . 예제\_4.3

- 형 변환 연산자를 이용한 예제

```
#include <stdio.h>

int main(void)
{
    int a = 5, b = 2;
    double re;

    re = a / b;           // 정수 연산의 결과는 정수
    printf("%.2f\n", re);

    re = (double)a / b;   // a를 형 변환
    printf("%.2f\n", re);

    re = (double)(a / b); // 나눗셈의 결과를 double형으로 변환
    printf("%.2f\n", re);

    return 0;
}
```



## 1. 예제

I . 예제\_5.1

- if문을 이용한 계산기 프로그램 예제

```
#include <stdio.h>

int main(void)
{
    int opt;
    double num1, num2;
    double result;

    printf("1.덧셈2.뺄셈3.곱셈4.나눗셈\n");
    printf("선택? ");
    scanf("%d", &opt);
    printf("두개의실수입력: ");
    scanf("%lf %lf", &num1, &num2);

    if(opt==1)
        result = num1 + num2;
    if(opt==2)
        result = num1 - num2;
    if(opt==3)
        result = num1 * num2;
    if(opt==4)
        result = num1 / num2;

    printf("결과: %f \n", result);

    return 0;
}
```

---

## II . 예제\_5.2

- if, else if, else를 이용한 성적 처리프로그램 예제

```
#include <stdio.h>

int main(void)
{
    int kor, eng, math;
    double ave;
    char grade;

    printf("[국어] [영어] [수학] 점수입력: ");
    scanf("%d %d %d", &kor, &eng, &math);
    fflush(stdin);

    ave = (double)(kor+eng+math)/3;

    if(ave>=90)
        grade = 'A';
    else if(ave>=80)
        grade = 'B';
    else if(ave>=70)
        grade = 'C';
    else if(ave>=60)
        grade = 'D';
    else
        grade = 'F';

    printf("평균은%2.f로%c등급입니다.\n", ave, grade);

    return 0;
}
```

---

### III. 예제\_5.3

- switch문을 이용한 등급 계산 프로그램 예제

```
#include <stdio.h>

int main(void)
{
    int score;
    char grade;
    int result;

    printf("점수입력: ");
    scanf("%d", &score);

    result = score / 10;

    switch(result)
    {
        case 10:
        case 9:
            grade = 'A';
            break;
        case 8:
            grade = 'B';
            break;
        case 7:
            grade = 'C';
            break;
        case 6:
            grade = 'D';
            break;
        default:
            grade = 'F';
    }

    printf("\n%d점=> %c등급\n", score, grade);

    return 0;
}
```

---

## 2. 문제

### I. 문제\_5.1

- 정수를 입력받아 아래와 같이 출력하는 프로그램을 작성(if문 이용)
  - 3의배수이면서, 4의배수에도 해당 => [ ]은(는) 3의 배수 이면서, 4의 배수입니다.
  - 3의배수에만 해당 => [ ]은(는) 3의 배수입니다.
  - 4의배수에만 해당 => [ ]은(는) 4의 배수입니다.
  - 3의배수도 4의배수도 해당안됨 => [ ]은(는) 3의 배수도 4의 배수도 아닙니다.
  - 0은 어떤 수의 배수도 아니기 때문에  
“0은(는) 3의 배수도 4의 배수도 아닙니다.” 라고 출력 되어 함

정수 입력 : 18

[18]은<는> 3의배수 입니다.

정수 입력 : 19

[19]은<는> 3의배수도 4의 배수도 아닙니다.

### II. 문제\_5.2

- 점수를 입력받아 등급을 출력하는 프로그램을 작성(switch문 이용)
  - 점수는 0~100점까지만 입력 가능하며, 그 외의 점수를 입력하면 “ 잘못된 입력 ”  
이라는 메시지 출력 후 종료
  - 90점 이상 : A등급
  - 80점 이상 : B등급
  - 70점 이상 : C등급
  - 60점 이상 : D등급
  - 60점 미만 : F등급

점수 입력 : 89

89점 ==> B등급

점수 입력 : 1000

잘못된 입력<0~100사이 입력>

점수 입력 : -10

잘못된 입력<0~100사이 입력>

## 1. 예제

## I . 예제\_6.1

- for문을 이용한 1~100까지의 누적합 출력 프로그램 예제

```
#include <stdio.h>

int main(void)
{
    int i, sum=0;

    for(i=1 ; i<=100 ; i++)
    {
        printf("%5d", i);
        sum += i;
    }

    printf("\n1부터 100까지의 합은 %d입니다!!\n", sum);

    return 0;
}
```

## II . 예제\_6.2

- for문을 이용한 1~1000사이 7의 배수의 합을 출력하는 프로그램 예제

```
#include <stdio.h>

int main(void)
{
    int i, sum=0;
    for(i=0 ; i<=1000 ; i+=7)
    {
        printf("%5d", i);
        sum += i;
    }
    printf("\n1~1000 사이의 7의 배수의 합은%d입니다!!\n", sum);

    return 0;
}
```

---

### III. 예제\_6.3

- for문을 이용한 시작값과 끝값 사이의 합계를 구하는 프로그램 예제

```
#include <stdio.h>

int main(void)
{
    int start, end, result;

    printf("시작값 끝값 입력: ");
    scanf("%d %d", &start, &end);

    for(result=0 ; start<=end ; start++)
    {
        result += start;
    }

    printf("시작과 끝 사이의 합계: %d \n", result);

    return 0;
}
```

### III. 예제\_6.4

- 다중 for문을 이용한 예제

```
#include <stdio.h>

int main(void)
{
    int i,j;

    for(i=1 ; i <=5 ; i++)
    {
        for(j=1 ; j<=i ; j++)
        {
            printf("*");
        }

        printf("\n");
    }

    return 0;
}
```

---

### III. 예제\_6.5

- 다중 for문을 이용한 예제

```
#include <stdio.h>

int main(void)
{
    int i,j;

    for(i=1 ; i <=9 ; i++)
    {
        if(i<=5)
        {
            for(j=1 ; j<=i ; j++)
                printf("*");

        }
        else
        {
            for(j=1 ; j<=10-i ; j++)
                printf("*");

        }
        printf("\n");
    }
    return 0;
}
```

## 2. 문제

### I. 문제\_6.1

- 다중 for문을 이용하여 아래와 같이 출력되는 프로그램 작성

```
*****
*****
****
***
**
*
계속하려면 아무 키나 누르십시오 . . .
```

---

## II . 문제\_6.2

- 다중 for문을 이용하여 구구단을 가로와 세로로 출력하는 프로그램 작성

```
## 구구단 가로출력 ##
2*1= 2  2*2= 4  2*3= 6  2*4= 8  2*5=10  2*6=12  2*7=14  2*8=16  2*9=18
3*1= 3  3*2= 6  3*3= 9  3*4=12  3*5=15  3*6=18  3*7=21  3*8=24  3*9=27
4*1= 4  4*2= 8  4*3=12  4*4=16  4*5=20  4*6=24  4*7=28  4*8=32  4*9=36
5*1= 5  5*2=10  5*3=15  5*4=20  5*5=25  5*6=30  5*7=35  5*8=40  5*9=45
6*1= 6  6*2=12  6*3=18  6*4=24  6*5=30  6*6=36  6*7=42  6*8=48  6*9=54
7*1= 7  7*2=14  7*3=21  7*4=28  7*5=35  7*6=42  7*7=49  7*8=56  7*9=63
8*1= 8  8*2=16  8*3=24  8*4=32  8*5=40  8*6=48  8*7=56  8*8=64  8*9=72
9*1= 9  9*2=18  9*3=27  9*4=36  9*5=45  9*6=54  9*7=63  9*8=72  9*9=81

## 구구단 세로출력 ##
2*1= 2  3*1= 3  4*1= 4  5*1= 5  6*1= 6  7*1= 7  8*1= 8  9*1= 9
2*2= 4  3*2= 6  4*2= 8  5*2=10  6*2=12  7*2=14  8*2=16  9*2=18
2*3= 6  3*3= 9  4*3=12  5*3=15  6*3=18  7*3=21  8*3=24  9*3=27
2*4= 8  3*4=12  4*4=16  5*4=20  6*4=24  7*4=28  8*4=32  9*4=36
2*5=10  3*5=15  4*5=20  5*5=25  6*5=30  7*5=35  8*5=40  9*5=45
2*6=12  3*6=18  4*6=24  5*6=30  6*6=36  7*6=42  8*6=48  9*6=54
2*7=14  3*7=21  4*7=28  5*7=35  6*7=42  7*7=49  8*7=56  9*7=63
2*8=16  3*8=24  4*8=32  5*8=40  6*8=48  7*8=56  8*8=64  9*8=72
2*9=18  3*9=27  4*9=36  5*9=45  6*9=54  7*9=63  8*9=72  9*9=81
계속하려면 아무 키나 누르십시오 . . .
```



## 1. 예제

### I . 예제\_7.1

- while문을 이용하여 입력받은 숫자만큼 화면에 문자열을 출력하는 예제

```
#include <stdio.h>

int main(void)
{
    int num;
    int i=0;

    printf("숫자입력: ");
    scanf("%d", &num);

    while(i<num)
    {
        printf("Hello World! \n");
        i++;
    }
    return 0;
}
```

### II . 예제\_7.2

- while문을 이용하여 숫자를 입력받아 그 수만큼 3의 배수를 출력하는 예제

```
#include <stdio.h>

int main(void)
{
    int i=0, num=0, cnt;

    printf("3의배수의개수: ");
    scanf("%d", &cnt);

    while(num++<cnt)
    {
        i+=3;
        printf("%d ", i);
    }
    return 0;
}
```

---

### III. 예제\_7.3

- do~while문을 이용하여 0과 100사이에 존재하는 짝수의 합을 출력하는 예제

```
#include <stdio.h>

int main(void)
{
    int total=0;
    int val=0;

    do
    {
        total+=val;
        val=val+2;
    }while(val<=100);

    printf("Total : %d \n", total);
    return 0;
}
```

### IV. 예제\_7.4

- goto문을 이용한 프로그램 예제

```
#include <stdio.h>

int main(void)
{
    int num;

    printf("숫자입력(1, 2, 3): ");
    scanf("%d", &num);

    if(num==1)
        goto label1;
    else if(num==2)
        goto label2;
    else
        goto label3;

    label1:
    printf("1을 입력하셨습니다! \n");
    return 0;

    label2:
    printf("2을 입력하셨습니다! \n");
}
```

```

return 0;

label3:
printf("3또는다른값을입력하셨습니다! \n");
return 0;
}

```

## 2. 문제

### I. 문제\_7.1

- 입력받은 분을 시간과 분으로 나눠 출력하는 프로그램 작성
- 프로그램은 종료되지 않고 무한반복 됨
- 분 입력이 0이면 프로그램 종료
- 분 입력이 0보다 작으면 다시 입력 받음

분 입력 <0 종료> : 100

[100분]은 [1시간 40분] 입니다.

분 입력 <0 종료> : 60

[60분]은 [1시간 0분] 입니다.

분 입력 <0 종료> : -10

0보다 작은수는 입력하실 수 없습니다.

분 입력 <0 종료> : 0

프로그램을 종료합니다.

### II. 문제\_7.2

- 입력받은 문자의 ASCII 코드를 출력하는 프로그램 작성
- 프로그램은 종료되지 않고 무한반복 됨
- 계속/종료 여부를 입력받아 S/s를 누르면 프로그램 종료

문자와 일치하는 ASCII 코드 검색

문자 입력 : a

문자 : a ==> 아스키 코드 : 97  
계속<아무키> / 종료<S/s>

문자 입력 : A

문자 : A ==> 아스키 코드 : 65  
계속<아무키> / 종료<S/s>

문자 입력 : I

문자 : I ==> 아스키 코드 : 73  
계속<아무키> / 종료<S/s> s

## 1. 예제

### I . 예제\_8.1

- 5층짜리 건물의 각 층 거주 인원을 입력 받아 건물의 거주인원 총합을 구하는 예제

```
#include <stdio.h>

int main(void)
{
    int arr[5];
    int total, i;

    for(i=0 ; i<5 ; i++)
    {
        printf("%d층에사는사람의수: ", i+1);
        scanf("%d", &arr[i]);
    }

    total = arr[0]+arr[1]+arr[2]+arr[3]+arr[4];
    printf("현재거주하는총인원: %d\n", total);
    return 0;
}
```

### II . 예제\_8.2

- 1차원 배열을 이용하여 최대 값을 구하는 예제

```
#include <stdio.h>

int main(void)
{
    int i, max;
    int score[5];

    for(i=0 ; i<5 ; i++)
    {
        printf("입력%d : ", i+1);
        scanf("%d", &score[i]);
    }

    max = score[0];
    for(i=1 ; i<5 ; i++)
```

---

```

    {
        if(max < score[i])
            max = score[i];
    }
    printf("\n최대값: %d\n", max);

    return 0;
}

```

### III . 예제\_8.3

- 문자열을 입력받아 문자열의 길이를 계산하는 예제

```

#include <stdio.h>

int main(void)
{
    char string[100];
    int len=0;

    printf("문자열입력: ");
    scanf("%s", string);

    while(string[len]!='\0')
        len++;

    printf("입력한문자열의길이는%d \n", len);
    return 0;
}

```

---

## 2. 문제

### I . 문제\_8.1

- 문자열을 입력받아 입력받은 문자열을 뒤집어 출력하는 프로그램 작성
- 배열의 크기는 100으로 함

```
문자열 입력: abcdef
변경된 문자열 : fedcba
계속하려면 아무 키나 누르십시오 . . .
```

### II . 문제\_8.2

- 문자열을 입력받아 입력받은 문자 중 ASCII코드값이 가장 큰 문자 출력하는 프로그램 작성
- 배열의 크기는 100으로 함

```
문자열 입력: 0135246abegjIEBKAN
아스키 코드값이 가장 큰 문자 : j
계속하려면 아무 키나 누르십시오 . . . █
```

## 1. 예제

I . 예제\_9.1

- 2차원 배열에 1~25까지의 숫자를 가로와 세로로 저장한 후, 출력하는 예제

```
#include <stdio.h>

int main(void)
{
    int arr[5][5];
    int i,j;
    int k = 1;

    for(i=0 ; i<5 ; i++)
    {
        for(j=0 ; j<5 ; j++)
        {
            arr[i][j] = k++;
            // arr[j][i] = k++; <-- 세로 출력
        }
    }

    for(i=0 ; i<5 ; i++)
    {
        for(j=0 ; j<5 ; j++)
        {
            printf("%5d", arr[i][j]);
        }
        printf("\n");
    }
    return 0;
}
```

---

## II. 예제\_8.2

- 2차원 배열을 이용하여 숫자를 짝수 행은 왼쪽부터 오른쪽으로 증가, 홀수 행은 오른쪽부터 왼쪽으로 증가하는 예제

```
#include <stdio.h>

int main(void)
{

    int arr[5][5];
    int i,j,k=1;

    for(i=0 ; i<5 ; i++)
    {
        if(i%2==0)
        {
            for(j=0 ; j<5 ; j++)
                arr[i][j] = k++;
        }
        else
        {
            for(j=4 ; j>=0 ; j--)
                arr[i][j] = k++;
        }
    }

    for(i=0 ; i<5 ; i++)
    {
        for(j=0 ; j<5 ; j++)
        {
            printf("%5d", arr[i][j]);
        }
        printf("\n");
    }

    return 0;
}
```



---

## 2. 문제

### I . 문제\_9.1

- 2차원 배열을 사용하여 구구단을 2~9단까지 출력하는 프로그램 작성

2	4	6	8	10	12	14	16	18
3	6	9	12	15	18	21	24	27
4	8	12	16	20	24	28	32	36
5	10	15	20	25	30	35	40	45
6	12	18	24	30	36	42	48	54
7	14	21	28	35	42	49	56	63
8	16	24	32	40	48	56	64	72
9	18	27	36	45	54	63	72	81
계속하려면 아무 키나 누르십시오 . . . █								

## 1. 예제

I . 예제\_10.1

- 사용자 정의 함수 예제

```
#include <stdio.h>

void func_line();      // line함수선언
void func_star();      // star함수선언

int main(void)
{
    func_star();        // star함수호출
    func_star();        // star함수호출
    func_line();        // line함수호출
    func_star();        // star함수호출
    func_line();        // line함수호출
    return 0;
}

void func_line()        // line함수정의
{
    printf("----\n");
}

void func_star()        // star함수정의
{
    printf("****\n");
}
```

---

## II . 예제\_10.2

### - 사용자 정의 함수 예제

```
#include <stdio.h>

void sum(int a, int b);
void div(int a, int b);

int main(void)
{
    int num1, num2;

    printf("입력1 : ");
    scanf("%d", &num1);

    printf("입력2 : ");
    scanf("%d", &num2);

    sum(num1, num2);
    div(num1, num2);

    return 0;
}

void sum(int a, int b)
{
    printf("%d + %d = %d\n", a, b, a+b);
}

void div(int a, int b)
{
    printf("%d / %d = %.2f\n", a, b, (double)a/b);
}
```

---

### III. 예제\_10.3

#### - 사용자 정의 함수 예제

```
#include <stdio.h>

int sum(int a, int b);
double div(int a, int b);

int main(void)
{
    int num1, num2, result;

    printf("입력1 : ");
    scanf("%d", &num1);

    printf("입력2 : ");
    scanf("%d", &num2);

    result = sum(num1, num2) * 10;

    printf("\n\n(%d + %d) * 10 = %d\n", num1, num2, result);
    printf("%d / %d = %.2f\n", num1, num2, div(num1, num2));

    return 0;
}

int sum(int a, int b)
{
    return a+b;
}

double div(int a, int b)
{
    double result;
    result = (double)a/b;

    return result;
}
```

---

## 2. 문제

### I. 문제\_10.1

- 사용자 정의 함수를 구현하여 코드를 완성 시켜라
- 이름 : power, 기능 : 정수의 제곱을 구해 리턴 하는 함수

```
#include <stdio.h>

int main(void)
{
    double re;
    int n;

    printf("제곱을 구할 숫자입력: ");
    scanf("%d", &n);

    re = power(n);
    printf("%d의 제곱은 %.0f입니다.\n", n, re);

    return 0;
}
```

### II. 문제\_10.2

- 사용자 정의 함수를 구현하여 코드를 완성 시켜라
- 이름 : sameNum함수, 기능 : 2개의 정수가 일치하면 1을 리턴 아니면 0을 리턴하는 함수

```
#include <stdio.h>

int main(void)
{
    int s1, s2;

    printf("두수입력:");
    scanf("%d %d", &s1, &s2);

    if(sameNum(s1, s2))
        printf("두수는 같습니다.\n");
    else
        printf("두수는 같지 않습니다.\n");

    return 0;
}
```

---

### III. 문제\_10.3

- 사용자 정의 함수를 구현하여 코드를 완성 시켜라
- 이름 : primeNum함수, 기능 : 소수라면 1을 리턴, 아니라면 0을 리턴하는 함수

```
#include <stdio.h>

int main(void)
{
    int su;

    printf("숫자입력:");
    scanf("%d", &su);

    if(primeNum(su))
        printf("%d는(은) 소수입니다.\n", su);
    else
        printf("%d는(은) 합성수입니다.\n", su);

    return 0;
}
```

### IV. 문제\_10.4

- 세 개의 수를 입력 받아 가장 큰 수와 가장 작은 수를 출력하는 프로그램 작성
- 가장 큰 수는 Max라는 이름의 사용자 정의함수로 구현
- 가장 작은 수는 Min이라는 이름의 사용자 정의함수로 구현

```
세 개의 정수 입력 : 10 5 59
가장 큰 수 : 59
가장 작은 수 : 10
계속하려면 아무 키나 누르십시오 . . .
```

### 1. 예제

I . 예제\_11.1

- &연산자를 이용하여 변수 주소 출력 예제

```
#include <stdio.h>

int main(void)
{
    char c = 'a';
    int i = 7;
    double d = 3.14;

    printf("c = %c\n", c);
    printf("i = %d\n", i);
    printf("d = %.2f\n", d);

    printf("c = %p\n", &c);
    printf("i = %p\n", &i);
    printf("d = %p\n", &d);

    return 0;
}
```

---

## II . 예제\_11.2

### - 일반변수의 크기와 포인터변수의 크기 비교 예제

```
#include <stdio.h>

int main(void)
{
    char c = 'a';
    int i = 7;
    double d = 3.14;

    // 포인터 변수 선언
    char *cp;
    int *ip;
    double *dp;

    // 변수의 주소를 포인터에 저장
    cp = &c;
    ip = &i;
    dp = &d;

    // 일반 변수의 크기 출력
    printf("i = %dbyte\n", sizeof(i));
    printf("c = %dbyte\n", sizeof(c));
    printf("d = %dbyte\n", sizeof(d));

    // 포인터 변수의 크기 출력
    printf("ip = %dbyte\n", sizeof(ip));
    printf("cp = %dbyte\n", sizeof(cp));
    printf("dp = %dbyte\n", sizeof(dp));

    return 0;
}
```



---

### III. 예제\_11.3

- \*연산자를 사용하여 메모리 간접참조 사용 예제

```
#include <stdio.h>

int main(void)
{
    char c = 'a';
    int i = 7;
    double d = 3.14;

    // 포인터변수선언
    char *cp = &c;
    int *ip = &i;
    double *dp = &d;

    // 포인터에 저장된 주소 출력
    printf("i = %p\n", ip);
    printf("c = %p\n", cp);
    printf("d = %p\n", dp);

    // 포인터가 가리키는 값 출력
    printf("ip = %d\n", *ip);
    printf("cp = %c\n", *cp);
    printf("dp = %.2f\n", *dp);

    return 0;
}
```

### IV. 예제\_11.4

- 포인터 연산으로 배열에 접근하는 예제

```
#include <stdio.h>

int main(void)
{
    int arr[3] = {1, 2, 3};
    int *ip;

    ip = arr; // 배열 이름은 배열의 시작주소
    printf("%p %p %p\n", ip+0, ip+1, ip+2);
    printf("%d %d %d\n", arr[0], arr[1], arr[2]);
    printf("%d %d %d\n", *(ip+0), *(ip+1), *(ip+2));
    return 0;
}
```

```
#include <stdio.h>

int main(void)
{
    int arr[5] = {1, 2, 3, 4, 5};
    int i, j = 1;
    int *ip;

    ip = arr;
    for(i=0 ; i<5 ; i++)
        printf("%d ", *ip++); // *ip와ip++이결합된형태
    printf("\n");

    ip = arr;
    for(i=0 ; i<5 ; i++)
        printf("%d ", ++*ip); // ++ip와*ip이결합된형태
    printf("\n");

    ip = arr;
    for(i=0 ; i<5 ; i++)
        printf("%d ", ++*ip); // ++arr[i]과같은형태
    printf("\n");

    // 위연산에의해값이변한배열값초기화
    for(i=0 ; i<5 ; i++)
        arr[i] = j;

    ip = arr;
    for(i=0 ; i<5 ; i++)
        printf("%d ", (*ip)++); // arr[i]++과같은형태
    printf("\n");

    return 0;
}
```

---

## 2. 문제

### 1. 문제\_11.1

- 크기가 5인 int형 배열 arr을 선언하고 1,2,3,4,5로 초기화한 다음, 포인터 p를 선언해서 배열 arr의 첫 번째 요소를 가리키게 함  
그 다음 포인터를 조작(포인터 연산을 의미함)해서 배열 요소의 값을 2씩 증가시킨 후, 전체 배열 요소를 출력하는 프로그램을 작성

3 4 5 6 7 계속하려면 아무 키나 누르십시오 . . . ■

### 1. 예제

#### I . 예제\_12.1

- 함수에 두 개의 정수를 전달한 후 전달한 두개의 정수를 교환하는  
함수 swapVal함수와 swapAdr함수 비교 예제

```
#include <stdio.h>

void swapVal(int x, int y);
void swapAdr(int *p1, int *p2);

int main(void)
{
    int a=3, b=5;

    swapVal(a,b);    // Call-by-Value
    printf("swapVal함수 호출 후 %d %d\n", a, b); // 값 바뀌지 않음

    swapAdr(&a, &b); // Call-by-Reference
    printf("swapAdr함수 호출 후 %d %d\n", a, b); // 값 바뀜

    return 0;
}

void swapVal(int x, int y)
{
    int temp;

    temp = x;
    x = y;
    y = temp;
}

void swapAdr(int *p1, int *p2)
{
    int temp;
    temp = *p1;
    *p1 = *p2;
    *p2 = temp;
}
```

---

## II . 예제\_12.2

- 배열에 저장된 데이터를 출력하는 함수 예제

```
#include <stdio.h>

void output(int *p);

int main(void)
{
    int arr[5] = {1, 2, 3, 4, 5};

    output(arr);

    return 0;
}

void output(int *p)
{
    int i;

    for(i=0 ; i<5 ; i++)
    {
        //printf("%d ", p[i]);
        printf("%d ", *(p+i));
    }

    printf("\n");
}
```

## 2. 문제

### I . 문제\_12.1

- 정수를 하나 입력받아 변수 a에 저장 후 입력받은 값의 제곱을 계산하여 다시 a에 대입해주는 프로그램 작성
- 입력받은 값의 제곱을 계산하는 함수(Square)를 독립적으로 구현
- main함수는 사용자 정의 함수를 호출하여 결과 값을 받아 출력

#### ① Call-by-Value 방식으로 구현

- square 함수 호출 시 변수 a를 전달
- 함수는 제곱 값을 계산하여 반환, 반환 값을 변수 a에 다시 저장
- 함수의 호출 형태 : a = Square(a);

```
정수 입력: 10
연산 결과 : 100
계속하려면 아무 키나 누르십시오 . . .
```

---

② Call-by-Reference 방식으로 구현

- square 함수 호출 시 변수 a의 주소 값을 전달
- 함수는 주소 값을 참조해서 변수 a의 값을 알아낸 다음 제곱을 계산하여 변수 a의 값을 변경
- 함수의 호출 형태 : Square(&a);

정수 입력: **10**

연산 결과 : **100**

계속하려면 아무 키나 누르십시오 . . . █

### 1. 예제

I . 예제\_13.1

- 문자열이 주소로 처리됨을 확인해 보기 위한 예제

```
#include <stdio.h>

void func(char *p);

int main(void)
{
    printf("%s\n", "Hello World!!");
    printf("%s\n", "Hello World!!"+2);
    printf("%s\n", "Hello World!!"+4);

    func("Hello Korea!!");

    return 0;
}

void func(char *p)
{
    printf("\n");

    printf("%s\n", p);
    printf("%s\n", p+2);
    printf("%s\n", p+4);
}
```

---

## II . 예제\_13.2

- 배열에 저장된 문자열을 대문자는 소문자로, 소문자는 대문자로 변환하는 프로그램 예제

```
#include <stdio.h>

void charConversion(char *str);

int main(void)
{
    char str[] = "Oh My God";

    printf("변환전출력: %s\n", str);
    charConversion(str);
    printf("변환후출력: %s\n", str);

    return 0;
}

void charConversion(char *str)
{
    int i = 0;

    while(str[i] != '\0')
    {
        if(str[i] >= 'A' && str[i] <= 'Z')
            str[i] += 32;
        else if(str[i] >= 'a' && str[i] <= 'z')
            str[i] -= 32;

        ++i;
    }
}
```



---

## 2. 문제

### I. 문제\_13.1

- 문자열을 입력받아 대문자를 모두 ‘\*’로 변환하여 출력하는 프로그램 작성

```
문자열 입력 : Hello World
```

```
입력 문자열 : Hello World
```

```
변경 문자열 : *ello *orld
```

### II. 문제\_13.2

- 문자열을 입력받아 입력된 알파벳의 개수만큼 별표를 찍는 프로그램 작성

```
문자열 입력 : hello world
```

```
=== 알파벳 개수 출력 ===
```

```
D : *
```

```
E : *
```

```
H : *
```

```
L : ****
```

```
O : **
```

```
R : *
```

```
W : *
```

---

## Chapter 14. 문자열 함수

---

### 1. 예제

#### I . 예제\_14.1

- 이름을 입력받아 이름의 길이를 구하는 예제

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

int main(void)
{
    char name[20];
    int nLen;

    printf("이름입력: ");
    gets(name);

    nLen = strlen(name);

    printf("글자수: %d\n", nLen);

    return 0;
}
```

#### II . 예제\_14.2

- 배열의 값을 다른 배열로 복사하는 예제

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

int main(void)
{
    char buf[255];
    char copy[255];
    printf("문자열입력: ");
    gets(buf);
    printf("\n입력문자열: %s\n", buf);
    // copy = buf; <-- 오류발생: 배열에배열대입불가능
    strcpy(copy, buf);
    printf("복사문자열: %s\n", copy);
    return 0;
}
```

---

### III. 예제\_14.3

- 입력받은 파일명에 ".txt"문자열을 연결하는 예제

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

int main(void)
{
    char str[80];
    printf("파일명입력: ");
    gets(str);
    strcat(str, ".txt");
    printf("\n파일명: %s\n", str);
    return 0;
}
```

### IV. 예제\_14.4

- 가위/바위/보를 선택한 후 문자열을 비교하여 조건에 따라 판별하는 예제

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

int main(void)
{
    char in[10];

    printf("가위/바위/보선택: ");
    gets(in);

    printf("\n컴퓨터보선택-> ");

    if(strcmp(in, "가위") == 0)
        printf("이겼습니다.\n");
    else if(strcmp(in, "바위") == 0)
        printf("졌습니다.\n");
    else if(strcmp(in, "보") == 0)
        printf("비겼습니다.\n");
    else
        printf("잘못선택하셨습니다.\n");
    return 0;
}
```

---

## V. 예제\_14.5

- atoi, atof를 이용하여 문자열을 정수와 실수형으로 변환하는 예제

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

int main(void)
{
    char s[10] = "1234", s1[10] = "-1234", s2[10] = "12 34"
    char st[10] = "12.34", st1[10] = "12.34ab", st2[10] = "abcd"

    printf("s : %d, s1 : %d, s2 : %d\n", atoi(s), atoi(s1), atoi(s2));
    printf("st : %.2f, st1 : %.2f, st2 : %.2f\n", atof(st), atof(st1),
    atof(st2));

    return 0;
}
```

## 2. 문제

### I. 문제\_14.1

- 문자열을 입력받아 입력받은 문자열 안에 존재하는 숫자들의 총 합을 계산

```
문자열 입력 : a12j1fa9a3g1ja01j3
숫자의 총 합 : 17
계속하려면 아무 키나 누르십시오 . . .
```

### 1. 예제

I . 예제\_15.1

- 배열의 크기를 매크로 상수로 선언한 뒤 for문을 매크로 상수를 이용하여 수정 예제

```
#include <stdio.h>

#define MAX 10

int main(void)
{
    int arr[MAX];
    int i;

    for(i=0 ; i<MAX ; i++)
        arr[i] = i + 1;

    for(i=0 ; i<MAX ; i++)
        printf("%d ", arr[i]);

    printf("\n");

    return 0;
}
```

---

## II . 예제\_15.2

- 매개변수에 괄호를 쓰지 않은 경우 매크로 함수의 문제점을 확인하기 위한 예제

```
#include <stdio.h>

#define ADD(x, y) x+y
#define MUL(x, y) x*y

int main(void)
{
    int result1, result2;

    result1 = 10 * ADD(3,4); // 10 * 3 + 4
    result2 = MUL(1+2, 3+4); // 1 + 2 * 3 + 4

    printf("result1 = %d\nresult2 = %d\n", result1, result2);

    return 0;
}
```

---

### III. 예제\_15.3

#### - 매크로 함수 연습 예제

```
#include <stdio.h>

#define OUT(s) printf(#s)
#define SUM1(a, b) printf("%d\n", a+b)
#define SUM2(a, b) printf("a+b=%d\n", a+b)
#define SUM3(a, b) printf("#a "+"#" #b "=%d\n", a+b)

int main(void)
{
    OUT>Hello World);
    printf("\n");

    OUT("Hello World");
    printf("\n");

    SUM1(3, 5);
    SUM2(3, 5);
    SUM3(3, 5);

    printf("3" "+" "5" "=%d\n", 3+5);

    return 0;
}
```

### 1. 예제

#### I . 예제\_16.1

- 학생 정보를 저장하는 student 구조체를 정의한 후 student형의 변수 선언
- 선언된 변수에 학생 정보를 입력 한 후 출력하는 프로그램 예제

```
#include <stdio.h>

struct student
{
    int no;
    char name[20];
    int score;
};

void linePrint(char ch, int cnt);

int main(void)
{
    struct student st1, st2;

    printf("\n\n\t\t***학생정보입력***\n\n");

    st1.no = 1;
    printf("학번: [%03d]\n", st1.no);

    printf("이름: ");
    fflush(stdin);
    gets(st1.name);

    printf("점수: ");
    fflush(stdin);
    scanf("%d", &st1.score);

    st2.no = 2;
    printf("\n학번: [%03d]\n", st2.no);

    printf("이름: ");
    fflush(stdin);
    gets(st2.name);
```



---

```
    printf("점수: ");
    fflush(stdin);
    scanf("%d", &st2.score);

    printf("\n\n\t\t***학생정보출력***\n\n");
    linePrint('=', 31);
    printf("%4s %-20s %5s\n", "학번", "이름", "점수");
    linePrint('-', 31);
    printf("%4d %-20s %5d\n", st1.no, st1.name, st1.score);
    printf("%4d %-20s %5d\n", st2.no, st2.name, st2.score);
    linePrint('=', 31);

    return 0;
}

void linePrint(char ch, int cnt)
{
    int i;
    for(i=0 ; i<cnt ; i++)
        printf("%c", ch);
    printf("\n");
}
```

---

## II . 예제\_16.2

- 16.1 예제를 구조체 배열을 이용하여 5명의 학생 정보를 저장하는 프로그램으로 수정하는 예제

```
#include <stdio.h>
#include <stdlib.h> // system();

#define STU_NUM 5

typedef struct student // typedef struct student student;
{
    int no;
    char name[20];
    int score;
}student;

void linePrint(char ch, int cnt);

int main(void)
{
    student s[STU_NUM];
    int i;

    for(i=0 ; i<STU_NUM ; i++)
    {
        system("cls");
        printf("\n\n\t\t***학생정보입력***\n\n");

        s[i].no = i+1;
        printf("학번: [%03d]\n", s[i].no);

        printf("이름: ");
        fflush(stdin);
        gets(s[i].name);

        printf("점수: ");
        fflush(stdin);
        scanf("%d", &s[i].score);

    }

    system("cls");
    printf("\n\n\t\t***학생정보출력***\n\n");
    linePrint('=', 31);
```

---

```
printf("%4s %-20s %5s\n", "학번", "이름", "점수");
linePrint('-', 31);

for(i=0 ; i<STU_NUM ; i++)
    printf("%4d %-20s %5d\n", s[i].no, s[i].name, s[i].score);

linePrint('=', 31);

return 0;
}

void linePrint(char ch, int cnt)
{
    int i;
    for(i=0 ; i<cnt ; i++)
        printf("%c", ch);
    printf("\n");
}
```

---

### III. 예제\_16.3

- 구조체 포인터로 구조체 배열로 선언된 멤버에 접근하는 예제

```
#include <stdio.h>

typedef struct score
{
    int kor, eng, mat, sum;
}score;

int main(void)
{
    int i;
    score s[3] = {
        {70, 70, 70},
        {80, 80, 80},
        {90, 90, 90}
    };

    score *p = s;

    for(i=0 ; i<3 ; i++)
    {
        s[i].sum = s[i].kor + s[i].eng + s[i].mat;
        printf("s[%d].sum = %d\n", i, s[i].sum);

        p[i].sum = p[i].kor + p[i].eng + p[i].mat;
        printf("p[%d].sum = %d\n", i, p[i].sum);

        (p+i)->sum = (p+i)->kor + (p+i)->eng + (p+i)->mat;
        printf("(p+%d)->sum = %d\n", i, (p+i)->sum);

        (*(p+i)).sum = (*(p+i)).kor + (*(p+i)).eng + (*(p+i)).mat;
        printf("(*(p+%d)).sum = %d\n", i, (*(p+i)).sum);

        printf("\n");
    }

    return 0;
}
```

### 1. 예제

I . 예제\_17.1

- 구조체 대입연산 예제

```
#include <stdio.h>

typedef struct data
{
    int a;
    double b;
    char c;
}DATA;

int main(void)
{
    DATA d1 = {50, 3.57, 'A'};
    DATA d2;

    d2 = d1;

    printf("%d, %.2f, %c\n", d1.a, d1.b, d1.c);
    printf("%d, %.2f, %c\n", d2.a, d2.b, d2.c);

    return 0;
}
```

---

## II . 예제\_17.2

### - 구조체 변수 Call-by-Value 예제

```
#include <stdio.h>

typedef struct data
{
    int a;
    double b;
    char c;
}DATA;

void input(DATA d);

int main(void)
{
    DATA d1 = {10, 3.14, 'A'};

    input(d1);
    printf("\n%d, %.2f, %c\n", d1.a, d1.b, d1.c);
    return 0;
}

void input(DATA d)
{
    printf("정수입력: ");
    scanf("%d", &d.a);
    fflush(stdin);

    printf("실수입력: ");
    scanf("%lf", &d.b);
    fflush(stdin);

    printf("문자입력: ");
    scanf("%c", &d.c);
}
```

---

### III. 예제\_17.3

- 구조체 변수 Call-by-Reference 예제

```
#include <stdio.h>

typedef struct data
{
    int a;
    double b;
    char c;
}DATA;

void input(DATA *p);

int main(void)
{
    DATA d1 = {10, 3.14, 'A'};

    input(&d1);
    printf("\n%d, %.2f, %c\n", d1.a, d1.b, d1.c);
    return 0;
}

void input(DATA *p)
{
    printf("정수입력: ");
    scanf("%d", &p->a);
    fflush(stdin);

    printf("실수입력: ");
    scanf("%lf", &p->b);
    fflush(stdin);

    printf("문자입력: ");
    scanf("%c", &p->c);
}
```

---

## 2. 문제

### I . 문제\_17.1

- 학생 한명의 정보를 student 구조체에 저장한 후 출력하는 프로그램 작성
  - 학생정보는 학번, 이름, 국어, 영어, 수학, 총점, 평균으로 구성
  - 학번은 1번으로 저장하며, 이름과 국/영/수 점수는 입력 받음
  - 입력받은 점수로 총점과 평균을 계산
  - 입력기능은 입력함수에서 출력기능은 출력함수에서 각각 구현

```
# 학생 정보 입력 #  
이름 : 이정수  
국어 : 80  
영어 : 90  
수학 : 100  
  
# 학생 정보 출력 #  
학번 이름      국어  영어  수학  총점   평균  
=====
```

1	이정수	80	90	100	270	90.00
---	-----	----	----	-----	-----	-------

```
계속하려면 아무 키나 누르십시오 . . .
```



---

```
#include <stdio.h>

typedef struct student
{
    /*
        여기에 구조체 멤버를 선언하시오.
    */
}student;

/*
    여기에 함수 원형을 선언하시오.
*/

int main(void)
{
    student s; // 구조체 변수선언

    // 입력 함수 호출
    // 출력 함수 호출

    return 0;
}

/*
    여기에 입력 함수를 정의하시오.
*/

/*
    여기에 출력 함수를 정의하시오.
*/
```

### 1. 예제

I . 예제\_18.1

- 포인터 배열 예제

```
#include <stdio.h>

int main(void)
{
    char *ptr[4] = {"Apple", "Banana", "Orange", "Strawberry"};
    int i, j;

    // 문자열 단위로 출력
    for(i=0 ; i<4 ; i++)
        printf("%s\n", ptr[i]);
    printf("\n");

    // 문자열을 문자 단위로 출력
    for(i=0 ; i<4 ; i++)
    {
        for(j=0 ; ptr[i][j]!='\0' ; j++)
            printf("%c", ptr[i][j]); // printf("%c", *(ptr[i]+j) );
        printf("\n");
    }

    return 0;
}
```

---

## II . 예제\_18.2

### - 더블포인터 예제

```
#include <stdio.h>

void output(char ** p);

int main(void)
{
    char *fruit[4] = {"apple", "banana", "orange", "strawberry"};
    output(fruit); // 배열의 이름은 배열의 시작주소(포인터의주소)

    return 0;
}

// 포인터 주소는 더블 포인터에 저장
void output(char ** p)
{
    int i;
    for(i=0 ; i<4 ; i++)
        printf("%s\n", p[i]); // printf("%s\n", *(p+i));
    printf("\n");
}
```

---

### III. 예제\_18.3

#### - 2차원 배열이름의 포인터 예제

```
#include <stdio.h>

int main(void)
{
    int a[6][4];
    int (*p)[4];
    int i, j, k=1;
    p = a;

    for(i=0 ; i<6 ; i++)
        for(j=0 ; j<4 ; j++)
            a[i][j] = k++;

    printf(" 1. %d\n", a[0]);
    printf(" 2. %d\n", *(a+0));
    printf(" 3. %d\n", (*(a+1))[0]);
    printf(" 4. %d\n", *(a[1]+2));
    printf(" 5. %d\n", *(*++p+3)+2);
    printf(" 6. %d\n", ++*(*(a+1)+2)+4);
    printf(" 7. %d\n", *(*p++ + 2)+4)+6);
    printf(" 8. %d\n", (*p[2]+3)+5);
    printf(" 9. %d\n", **(p+1)+3);
    printf("10. %d\n", *(p[0]+6)+5);
}
```

---

## Chapter 19. 파일 입 · 출력

---

### 1. 예제

I . 예제\_19.1

- 입력받은 한 글자를 스트림에 출력하는 프로그램 예제

```
#include <stdio.h>

int main(void)
{
    FILE *fp;
    char ch;

    fp = fopen("out.txt", "wt"); // 파일생성 - 쓰기모드
    if(fp == NULL)
    {
        printf("File Open Fail\n");
        return 0;
    }

    printf("문자입력: ");
    scanf("%c", &ch);

    fputc(ch, fp);

    return 0;
}
```

---

## II . 예제\_19.2

- ctrl+z를 누르기 이전까지 문자열을 입력 받은 후 스트림에 출력하는 프로그램 예제

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

void input(char *str);

int main(void)
{
    FILE *fp;
    char str[255];

    fp = fopen("out.txt", "wt");

    if(fp == NULL)
    {
        printf("File Open Fail\n");
        return 0;
    }

    input(str);
    fputs(str, fp);
    printf("out.txt 파일에 저장완료\n");
    fclose(fp);

    return 0;
}

void input(char *str)
{
    char temp;
    int i=0;

    printf("문자열입력(종료- Ctrl + Z)\n");
    while(temp = getchar())
    {
        if(temp == EOF) // Ctrl+Z
            break

        str[i++] = temp;
    }
    str[i] = '\0'
}
```

---

### III. 예제\_19.3

- 저장할 파일명을 입력받아 파일을 복사하는 프로그램 예제
- 프로젝트(파일이 읽고 쓰이는 위치)에 'test.txt'파일을 미리 만들어 준비해야 함

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

void fileCopy(FILE *fp, FILE *fp2);
void fileNameInput(char *fileName);

int main(void)
{
    FILE *fp, *fp2;
    char fileName[255];

    fileNameInput(fileName);

    fp = fopen("test.txt", "rt");
    fp2 = fopen(fileName, "wt");

    if(fp == NULL || fp2 == NULL)
    {
        printf("FILE Open Fail\n");
        return 0;
    }

    fileCopy(fp, fp2);
    printf("%s 파일이 복사되었습니다.\n", fileName);

    fcloseall();

    return 0;
}

void fileNameInput(char *fileName)
{
    printf("저장파일명입력: ");
    gets(fileName);
    strcat(fileName, ".txt");
}

void fileCopy(FILE *fp, FILE *fp2)
{
    char temp[255];
```

---

```
char *pSave;

while(pSave = fgets(temp, 255, fp))
{
    if(pSave == NULL)
        break

    fputs(temp, fp2);
}
}
```



## 1. 예제

I . 예제\_20.1

- 학생 수를 입력받아 입력받은 학생 수만큼 수학점수를 저장할 수 있는 메모리를 할당, 할당 된 메모리에 점수를 입력받아 평균을 구하는 프로그램 예제

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

int main(void)
{
    int i, in, sum=0;
    int *p;

    printf("학생수입력: ");
    scanf("%d", &in);

    p = (int *)malloc(in*sizeof(int));

    printf("\n");
    for(i=0 ; i<in ; i++)
    {
        printf("%d번수학점수: ", i+1);
        scanf("%d", &p[i]);
        sum += p[i];
    }

    printf("\n학생수: %d명\n평균: %.2f\n", in, (double)sum/in);

    free(p);

    return 0;
}
```

---

## II . 예제\_20.2

- 학생 수를 입력받은 학생 수만큼 구조체를 할당하여 입출력하는 프로그램 예제

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

typedef struct Student
{
    int kor, eng, mat, sum;
}Student;

void input(Student *pSt, int in);
void output(Student *pSt, int in);

int main(void)
{
    int in;
    Student *pSt;

    printf("학생수입력: ");
    scanf("%d", &in);

    pSt = (Student *)malloc(in * sizeof(Student));

    input(pSt, in);
    output(pSt, in);

    free(pSt);

    return 0;
}

void input(Student *pSt, int in)
{
    int i;
    for(i=0 ; i<in ; i++)
    {
        printf("\n국어점수: ");
        scanf("%d", &pSt[i].kor);

        printf("영어점수: ");
        scanf("%d", &pSt[i].eng);

        printf("수학점수: ");
```

---

```
        scanf("%d", &pSt[i].mat);

        pSt[i].sum = pSt[i].kor + pSt[i].eng + pSt[i].mat;
    }
}

void output(Student *pSt, int in)
{
    int i;
    printf("\n\n ***** 점수출력*****\n\n");
    printf("%4s %4s %4s %4s %4s\n", "번호", "국어", "영어", "수학",
"총점");

    for(i=0 ; i<in ; i++)
        printf("%4d %4d %4d %4d %4d\n", i+1, pSt[i].kor, pSt[i].eng,
pSt[i].mat, pSt[i].sum);
}
```

---

## II . 예제\_20.3

- 구조체 멤버의 문자열을 동적 할당하여 출력 하는 프로그램 예제

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

typedef struct employee
{
    char *name;
    char *phone;
    char *addr;
}emp;

int main(void)
{
    int i, in;
    char str[255];
    emp *np;

    printf("사원수입력: ");
    scanf("%d", &in);
    fflush(stdin);

    np = (emp*)malloc(sizeof(emp)*in);

    for(i=0 ; i<in ; i++)
    {
        printf("\n사원번호: [%04d]\n", i+1);

        printf("사원명: ");
        gets(str);
        np[i].name = (char*)malloc(strlen(str)+1);
        strcpy(np[i].name, str);

        printf("전화번호: ");
        gets(str);
        np[i].phone = (char*)malloc(strlen(str)+1);
        strcpy(np[i].phone, str);

        printf("주 소: ");
        gets(str);
        np[i].addr = (char*)malloc(strlen(str)+1);
        strcpy(np[i].addr, str);
    }
}
```

---

```
    }

    printf("\n\n%-8s %-10s %-15s %-30s\n", "사원번호", "사원명",
"전화번호", "주소");

    for(i=0 ; i<in ; i++)
    {
        printf("%08d %-10s %-15s %-30s\n", i+1, np[i].name,
np[i].phone, np[i].addr);
    }

    for(i=0 ; i<in ; i++)
    {
        free(np[i].name);
        free(np[i].phone);
        free(np[i].addr);
    }
    free(np);
}
```