

1. 아래와 같은 조건을 적용하여 미니계산기를 작성하세요

① 다음과 같이 출력될 수 있도록 작성하세요

```
*****미니계산기*****
1.더하기
2.곱하기
3.빼기
4.나누기
연산자를 선택하세요.
```

② 다음과 같이 연산자를 선택하고 두 수를 입력하면 결과가 출력될 수 있도록 작성하세요

```
*****미니계산기*****
1.더하기
2.곱하기
3.빼기
4.나누기
연산자를 선택하세요.1
숫자를 입력하세요 100 100
계산결과=>200
```

2. 아래와 같은 화면이 나타나고 조작하면 결과가 출력될 수 있도록 작성합니다.

```
-----Menu-----
1. 가정용 (liter당 50원)
2. 상업용 (liter당 45원)
3. 공업용 (liter당 30원)
-----
메뉴를 선택하세요=>
-----
```

위의 화면에서 메뉴를 선택 후 사용량을 입력 받습니다.

```
-----Menu-----
1. 가정용 (liter당 50원)
2. 상업용 (liter당 45원)
3. 공업용 (liter당 30원)
-----
메뉴를 선택하세요=>
-----
1
사용량을 입력하세요=>
200
```

다음과 같은 결과가 나타나도록 작성하세요.

사용자 코드, 사용량, 총수도요금

cf) 사용요금 = 사용량 * 리터당 가격

총수도요금 = 수도사용요금 + 세금(요금의 5%)

```
-----Menu-----
1. 가정용 (liter당 50원)
2. 상업용 (liter당 45원)
3. 공업용 (liter당 30원)
-----
메뉴를 선택하세요=>
-----
1
사용량을 입력하세요=>
200
=====
사용자 코드: 1
사용 요금: 10000
총수도 요금: 10500.0
=====
```

3. 1 부터 1000 까지의 홀수의 합, 짝수의 합, 총합을 출력하세요

<출력형태>

1 부터 1000 까지의 총합_____

1 부터 1000 까지 홀수의 합_____

1 부터 1000 까지 짝수의 합_____

4. 1 보다 크고 10 보다 작은 정수를 입력받아서 아래와 같은 실행

결과가 나타나도록 작성하세요. 입력받은 숫자에 해당하는 구구단을 완성시키는 프로그램 입니다. 2 보다 작거나 9 보다 큰 정수가 입력될 경우에는 "잘못된 숫자가 입력되었습니다."라는 경고 메 시지를 출력하고 프로그램을 종료합니다.

```
1 보다 크고 10 보다 작은 정수를 입력하세요. : 3
```

```
3 * 1 = 3
```

```
3 * 2 = 6
```

```
3 * 3 = 9
```

```
...
```

```
3 * 9 = 27
```

5. 초기 값, 마지막 값, 증가분을 입력 받아서, 초기값부터 마지막 값까지, 증가분씩 값을 증가시키면서 각 값들의 총합을 구하는 프로그램을 작성하세요. 단, 합을 구한 결과가 1000 을 넘 으면, 결과에 2000 을 한번 더 더하여 출력합니다.

실행 예 1)

```
초기값을 정수로 입력하세요. : 1 마지막  
값을 정수로 입력하세요. : 10
```

```
증가분을 정수로 입력하세요. : 1  
총합은 55 입니다.
```

실행 예 2)

```
초기값을 정수로 입력하세요. : 10  
마지막값을 정수로 입력하세요. : 100
```

```
증가분을 정수로 입력하세요. : 5  
총합은 3045 입니다.
```

6. 소수란 자신과 1 외의 다른 수로는 나누어 떨어지지 않는 정수입니다. 아래와 같이 2~100 사이의 숫자를 입력하면 소수인지 여부를 판별하는 프로그램을 작성하십시오.

실행 예 1)

2 ~ 100 사이의 정수를 입력하세요. : 47
47 는(은) 소수입니다.

실행 예 2)

2 ~ 100 사이의 정수를 입력하세요. : 93
93 는(은) 소수가 아닙니다.

* 참고 : 2 부터 100 사이의 소수는 다음과 같습니다. 프로그램 테스트에 참고하기 바랍니다.

2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43, 47, 53, 59, 61,
67, 71, 73, 79, 83, 89, 97

7. 1 부터 50 까지의 숫자 중 소수가 모두 출력되도록 작성하세요

2
3
5
7

..... 중간생략.....

41
43
47

8. 고전 암호학(Classical Cryptography)에서 사용하는 기법 중 "Caesar Cipher" 는 암호화의 대상이 되는 각각의 문자들을 알파벳 상의 세번째 오른쪽 문자로 치환하는 간단한 치환 암호기 법(즉, a 는 d 로, b 는 e 로, ... , x 는 a 로, y 는 b 로, z 는 c 로)입니다. "everyday we have is one more than we deserve" 라는 문자열에 대해 Caesar Cipher 를 적용하여 암호화된 문자열 값을 아래와 같이 출력하도록 완성하십시오. (단, 공백에 대해서는 치환을 적용하지 않습니다.)

암호화할 문자열 : everyday we have is one more than we deserve

암호화된 문자열 : hyhubgdb zh kdyh lv rqh pruh wkdq zh ghvhuyh

9. 두 숫자의 최대 공약수를 구하는 프로그램을 작성하십시오.

최대 공약수는 두 숫자를 공통된 숫자로 나누어도 나머지가 0 이 되는 가장 큰 수 를 의미합니다. 예를 들어 2 와 5 의 최대 공약수는 1 입니다. 왜냐하면, 1 부터 2 까 지의 숫자중 공통으로 나눌 수 있는 숫자는 1 뿐이기 때문입니다. 다른 예로, 5 와 15 의 최대 공약수는 5 가 됩니다. 메소드의 이름은 gcd 로 하며, 매개변수는 두개 의 정수를 받으며, 리턴 하는 타입 역시 정수입니다.

[입력값]

2, 5

5,15

250,30

[실행결과]

1

5

10

10. 임의의 정수를 입력 받아서, 1 부터 1,000 까지의 수 중에서 입력받은 정수의 배수의 개수와 배수들의 합을 계산하십시오.

[출력형태]

양의 정수를 입력하세요 : 7

배수갯수 = 142

7 의 배수의 합 = 71071

11. 정수를 한 개 입력받아, 1 부터 입력받은 수까지 각각에 대해 제곱을 구해 프린트하는 프로그램을 작성해 보세요. 단, while 문을 사용하세요.

예 1

입력:

3

출력:

1 1
2 4
3 9

예 2

입력:

5

출력:

1 1
2 4
3 9
4 16
5 25

12. input()으로 사용자로부터 입력받은 정수를 계속 더해나가다가, 음수가 입력되면 중단하고 그 전까지 계산한 값을 출력할 수 있도록 작성하세요.

예

입력:

```
1
2
3
-1
```

출력:

```
6
```

예 2

입력:

```
50
60
70
-100
```

출력:

```
180
```

13. 주어진 년도의 월에 해당하는 말일을 계산하도록 작성하세요. 윤년인 경우에는 다음과 같은 조건에 의해 처리합니다.

[윤년의 조건]

- 1) 4의 배수인 해는 윤년.
- 2) 4의 배수이면서 100의 배수인 해는 윤년이 아님.
- 3) 100의 배수이면서 400의 배수인 해는 윤년.

[실행 결과]

```
2010년 2월의 말일은 28일 입니다
```