

## <C프로그래밍 및 실습> 5차 과제 (8장 함수)

### ※ 문제에 대한 안내

- 입출력 예시에서 □는 공백 문자를 의미한다.
- 입출력 예시에서 ↳ 이 후는 각 입력과 출력에 대한 설명이다.
- OJ에서 Sample Submit 기능 사용가능한 문제들입니다.
- 문제지 및 문제의 커뮤니티, 개인 블로그 등 인터넷 사이트 게시를 절대 금지합니다.

[ 문제 1-1 ] 암호화된 문자의 원문자를 계산하여 출력하는 프로그램을 작성하시오.

가장 간단한 문자 암호기법 중 하나는 각 문자를 (알파벳 순서로) 암호키 M 만큼 뒤에 있는 문자로 변경시키는 것이다. 예를 들어, 암호키 M=3일 때, 문자 a는 알파벳 순서에서 3번째 뒤에 있는 d로 변경하고, b → e, c → f 와 같은 식으로 변경하고, 알파벳 순서에서 문자 z 뒤에는 문자 a가 연결되어 있다고 가정하여 w → z, x → a, y → b, z → c 로 변경한다.

문자 10개를 입력받아 배열에 저장하고, 정수 M을 입력 받는다. 입력된 문자는 원문자에 암호키 M으로 암호화된 문자이다. 입력된 문자의 원문자를 출력하는 프로그램을 decrypt 함수를 사용하여 작성하시오. 단, 영어대문자와 영어소문자를 제외한 문자는 암호화하지 않는다.

- decrypt함수

- 함수 원형: **char decrypt(char ch, int M)**
- 인자: 문자 ch와 정수 M
- 암호키 M으로 암호화된 문자인 ch의 원문자를 계산한다.
- 반환 값 : 암호키 M으로 암호화된 문자인 ch의 원문자를 return. 만약, ch가 영어소문자나 영어대문자가 아닌 경우에는 ch를 return.

▶ main 함수 위에 함수원형선언을 하고, 함수 본체를 main 함수 아래에 작성 하시오.

입력 예시 1

출력 예시 1

lveocmboox 10	↳ 암호화된 문자	bluescreen	↳ 원문자
------------------	-----------	------------	-------

[ 문제 1-2 ] 암호화된 문자의 원문자를 계산하여 출력하고 다시 암호화하여 출력하는 프로그램을 작성하시오.

종료조건까지 문자를 연속해서 입력받아 배열에 저장하고, 정수 M1과 M2를 입력 받는다. 입력된 문자는 원문자에 암호키 M1으로 암호화된 문자이다. 입력된 문자의 원문자를 계산하여 출력한 후, 암호키 M2로 다시 암호화하여 출력하는 프로그램을 작성 하시오. 단, 영어대문자와 영어소문

자를 **제외한** 문자는 **암호화하지 않는다**.

- 종료조건 : '\*' 입력
- 입력된 문자의 최대 길이는 100 이다.

- decrypt함수

- 함수 원형: **char decrypt(char ch, int M)**
- 인자: 문자 ch와 정수 M
- 암호키 M으로 암호화된 문자인 ch의 원문자를 계산한다.
- 반환 값 : 암호키 M으로 암호화된 문자인 ch의 원문자를 return. 만약, ch가 영어소문자나 영어대문자가 아닌 경우에는 ch를 return.

- encrypt함수

- 함수 원형: **char encrypt(char ch, int M)**
- 인자: 문자 ch와 정수 M
- 문자 ch를 암호키 M으로 암호화 한다.
- 반환 값 : ch를 암호키 M으로 암호화한 문자를 return. 만약, ch가 영어소문자나 영어대문자가 아닌 경우에는 ch를 return.

▶ main 함수 위에 함수원형선언을 하고, 함수 본체를 main 함수 아래에 작성 하시오.

입력 예시 1

출력 예시 1

lveocmbox*	↪ 암호화된 문자	bluescreen	↪ 원문자
10 20	↪ M1 M2	vfoymwlyyh	↪ 다시 암호화

입력 예시 2

출력 예시 2

eoxh VFUHHQ #57*	blue SCREEN #57
29 39	oyhr FPERRA #57

[ 문제 2-1 ] 0이 입력될 때까지 양의정수 N개를 입력받아 배열 A에 저장하고,, 두 개의 정수 S와 E를 입력받아 (S, E < N, S<E) A[S]에서 A[E]까지의 배열을 A[S]는 A[E]와 교환하고 A[S+1]은 A[E-1]과 교환하는 과정을 반복하는 프로그램을 작성하시오. 또한, A[S]에서 A[E]까지의 배열에서 최대값과 최소값을 구하는 함수 max\_num, min\_num 함수를 작성하시오.

- max\_num, min\_num함수

- 함수 원형: **int max\_num(int N1, N2), int min\_num(int N1, N2)**
- 인자: 정수 N1, N2
- 두 정수의 최대값, 최소값을 계산
- 반환 값 : 두 정수의 최대값, 최소값을 return

입력 예시 1

출력 예시 1

12 13 10 92 2 34 32 41 51 100 0	12 13 10 41 32 34 2 92 51 100
3 7	92 2

[ 문제 2-2 ] 0이 입력될 때까지 양의정수 N개를 입력받아 배열 A에 저장하고, 두 개의 정수 S와 E를 T개 입력받아 [문제1-1]의 과정을 T번 반복하는 프로그램을 작성하시오. 또한, 각 반복에서 주어진 배열에서 최대값과 최소값을 구하는 함수 max\_num, min\_num 함수를 [문제 2-1]에서 주어진 정의를 사용하여 작성하시오.

입력 예시 1

출력 예시 1

3 81 9 12 4 36 33 91 10 1 0	3 81 9 91 33 36 4 12 10 1
3 ↦ T= 3 개	91 4
3 7 4 5 0 4	3 81 9 91 36 33 4 12 10 1
	36 33
	36 91 9 81 3 33 4 12 10 1
	91 3

1) A[3]에서 A[7]까지 A[3]->A[7], A[4]->A[6] : 3 81 9 91 33 36 4 12 10 1

최대값과 최소값 : 91 4

2) 1)의 결과에서 A[4]와 A[5]를 교환 : 3 81 9 91 36 33 4 12 10 1

최대값과 최소값 : 36 33

3) 2)의 결과에서 A[0]에서 A[4]까지의 값을 1)과 동일하게 반복 : 36 91 9 81 3 33 4 12 10 1

최대값과 최소값 : 91 3

[ 문제 3 ] 세 정수 N, M, K를 입력받아 N부터 M까지의 수 중에서

(1) K의 배수 이면, 해당 수의 각 자릿수 중 최대값을 구하고

(2) K의 배수가 아니면, K로 나눈 나머지 중에서 최대값을 구한 후

(1) 과 (2) 중에서 제일 큰 수를 출력하시오.

-multiple, maximum, digit\_maximum 세 개의 함수를 사용하여 작성하시오.

1 < N < M < 1000 이고, 1 < K < 1000 이다.

- multiple 함수의 정의 및 사용

○ 인자: 정수 x와 정수 y

○ 반환형: int

○ 반환값: 정수 x가 정수 y의 배수이면 1, 아니면 0을 return

- maximum 함수의 정의 및 사용

○ 인자: 정수 x와 정수 y

○ 반환형: int

○ 반환값: 정수 x와 정수 y 중 큰 수를 return

- digit\_maximum 함수의 정의 및 사용

○ 인자는 정수 x

○ 반환형: int

○ 반환값: 정수 x의 각 자릿수를 계산하고 maximum 함수를 이용하여 그 중 가장 큰 수를

계산하고 return

입력 예시 1

출력 예시 1

12 18 3	8
---------	---

12, 13, ..., 17, 18 중에서 3의 배수는 12, 15, 18 이며, 이 중에서 각 자릿수 최대값은 8 이다.  
 12, 13, ..., 18 에서 3으로 나눈 나머지는 0, 1, 2, 0, 1, 2, 0 이며, 이 중에서 최대값은 2 이다.  
 8과 2중에서 최대값은 8 이다.

입력 예시 2

출력 예시 2

9 18 11	10
---------	----

9, 10, ..., 17, 18 중에서 11의 배수는 11 이며, 이 중에서 각 자릿수 최대값은 1 이다.  
 11로 나눈 나머지는, 9, 10, 0, 1, ..., 7 이며, 이 중에서 최대값은 10 이다.  
 1과 10중에서 최대값은 10 이다.

**[ 문제 4-1 ]** 양의 정수 N과 M을 입력받고, N개의 양의 정수를 입력받아 이 중 M번째 소수까지만 출력하는 프로그램을 is\_prime\_number() 함수를 사용하여 작성하시오. 만일, 입력된 정수 중 소수의 개수가 M개 보다 적으면, 입력된 소수를 모두 출력하고, 소수가 하나도 없으면 0을 출력한다.

- is\_prime\_number 함수

- 함수 원형: **int is\_prime\_number(int num)**
- 양의 정수 num이 소수이면 1을 반환하고, 소수가 아니면 0을 반환한다.

입력 예시 1

출력 예시 1

10 3 1 2 3 4 5 6 7 8 9 10	↪ 정수 10개를 읽어 소수 3개 출력 □ 2 3 5
------------------------------	----------------------------------

입력 예시 2

출력 예시 2

10 5 1 2 3 4 5 6 7 8 9 10	□ 2 3 5 7
------------------------------	-----------

입력 예시 3

출력 예시 3

10 5 32 33 34 35 36 42 44 45 46 54	0
---------------------------------------	---

**[ 문제 4-2 ]** 양의 정수 N과 N개의 양의 정수를 입력받아 연속적으로 소수가 나온 최대 횟수를 카운트하고 그 구간의 마지막 소수를 출력 하는 프로그램을 count\_prime\_number() 함수와 is\_prime\_number() 함수를 이용하여 작성하시오. 단, 100 이상인 수는 연속적인 소수 횟수의 카운트에서 제외 한다(소수로 생각하지 않음). 입력된 정수 중 소수가 하나도 없으면 0을 출력한다. 소수연속구간 최대 회수를 출력하고, 다음 줄에 그 구간의 소수들 중 가장 마지막 소수를 출력한다.

- count\_prime\_number 함수

- 함수 원형: **int count\_prime\_number(int cnt, int num)**
  - 연속된 소수의 개수를 세는 cnt 값과 양의 정수 num을 전달 받아, num이 소수이면, cnt+1한 값을 반환하고, num이 소수가 아니거나 100이상의 정수이면, 0을 반환한다.
  - num 값이 소수인지 확인하기 위해 is\_prime\_number() 함수를 호출한다.
- is\_prime\_number 함수
- 함수 원형: **int is\_prime\_number(int num)**
  - 양의 정수 num이 소수이면 1을 반환하고, 소수가 아니면 0을 반환한다.
- 소수가 6번 연속 나온 구간이 제일 길면 6을 출력한다. (입력 예시 1)
- 입력 중간에 100이상의 소수가 나와도 100이상의 수이므로 소수 구간은 새로 시작해야한다. (입력 예시 2)
- 한 번도 소수가 없었다면 0을 출력한다. (입력 예시 3)
- 최대 길이가 같은 구간이 있으면 제일 처음 나온 구간의 정보를 출력한다.

입력 예시 1

15
11 13 17 19 71 <b>73</b> 74 2 3 5 58 97 89 79 47

출력 예시 1

6 ↦ 11 13 17 19 71 73
73 ↦ 그 구간의 마지막 소수

입력 예시 2

17
11 13 17 19 101 71 73 74 2 3 5 58 <u>97 89 79 11 37</u>

출력 예시 2

5 ↦ 97 89 79 11 37
37 ↦ 그 구간의 마지막 소수

입력 예시 3

10
32 33 34 35 36 42 44 45 46 54

출력 예시 3

0
---

**[ 문제 5-1 ]** 양의 정수 N을 입력 받아 N의 각 자릿수의 합을 구하고, N과 합을 출력하는 프로그램을 add\_digits 함수를 사용하여 작성하시오.

- add\_digits함수
- 함수 원형: **int add\_digits(int N)**
  - 인자: 정수 N
  - 정수 N의 각 자리수의 합을 계산
  - 반환 값 : 정수 N의 각 자리수의 합을 return

입력 예시 1

12345

출력 예시 1

12345 15

입력 예시 2

24680

출력 예시 2

24680 20

**[ 문제 5-2 ]** 10 이상의 양의 정수 N을 입력 받아 디지털근을 구하여 출력하는 프로그램을 위 문제에서 정의한 add\_digits 함수를 사용하여 작성하시오.

- 디지털근, 예)  $6234 \rightarrow 6+2+3+4 = 15 \rightarrow 1+5 = 6$

입력 예시 1

6234

출력 예시 1

6

입력 예시 2

123456789

출력 예시 2

9

**[ 문제 5-3 ]** 종료 조건 시까지 0 혹은 양의 정수 N을 반복해서 입력 받아, 입력받은 정수 중 디지털근의 값이 가장 작은 정수와 해당 정수의 디지털근의 값을 출력하는 프로그램을 위 문제에서 정의된 add\_digits 함수를 이용하여 작성하시오.

디지털 근의 값이 같은 정수가 2개 이상인 경우 먼저 입력된 수를 우선으로 출력한다.

- 종료 조건 : 음수 입력 (단, 첫 번째 입력된 정수는 음수가 아니라고 가정한다.)

입력 예시 1

1012 24 135 7 -5

출력 예시 1

1012 4

입력 예시 2

912 175 1982 -4

출력 예시 2

1982 2

### 제출기한 및 방법

\* OJ시스템(<https://ex-oj.sejong.ac.kr/index.php/auth/login>) 내의 **과제5**를 이용하여 제출