

## 2021-1 프로그래밍언어 실습 3

### 3.1 지정된 연월일이 서기 1년 1월 1일로부터 몇 번째 날인지, 그리고 무슨 요일인지를 계산하여 출력하는 프로그램 작성

1) 연월일 3개 정수로 지정된 날짜를 입력 받아, 서기 (AD) 1년 1월 1일로부터 몇 번째 날이며, 무슨 요일인가를 계산하여 출력하기 위한 함수 `bool isLeapYear(int y)`와 `int getDaysFromJan01AD01(int year, int month, int day)`를 작성하라.

참고로 서기 1년 1월 1일은 월요일이다.

2) 오늘의 연 월 일 3개 정수를 입력 받아 서기 (AD) 1년 1월 1일로부터 몇 번째 날이며, 무슨 요일인가를 계산하여 출력하라. 달에 해당하는 이름을 출력하고, 요일에 해당하는 이름을 각각 출력할 것.

3) 2011년 이후 올해까지의 매년 1월 1일이 서기 1년 1일로부터 몇 번째 날인지, 그리고 무슨 요일인가를 출력하라.

4) 실행 예제:

```
Input year, month, day to check : 2021 3 13
The week day of March 13, 2021 is 737862-th day from Jan 1, 1 (SAT)

The week day of January 1, 2011 is 734138-th day from Jan 1, 1 (SAT)
The week day of January 1, 2012 is 734503-th day from Jan 1, 1 (SUN)
The week day of January 1, 2013 is 734869-th day from Jan 1, 1 (TUE)
The week day of January 1, 2014 is 735234-th day from Jan 1, 1 (WED)
The week day of January 1, 2015 is 735599-th day from Jan 1, 1 (THR)
The week day of January 1, 2016 is 735964-th day from Jan 1, 1 (FRI)
The week day of January 1, 2017 is 736330-th day from Jan 1, 1 (SUN)
The week day of January 1, 2018 is 736695-th day from Jan 1, 1 (MON)
The week day of January 1, 2019 is 737060-th day from Jan 1, 1 (TUE)
The week day of January 1, 2020 is 737425-th day from Jan 1, 1 (WED)
The week day of January 1, 2021 is 737791-th day from Jan 1, 1 (FRI)
```

### 3.2 5 × 5 크기의 행렬 A, B에 대한 덧셈, 뺄셈, 곱셈 연산

- 1) 2개의 5 × 5 크기의 행렬 A, B의 덧셈, 뺄셈, 곱셈을 계산하여 그 결과를 C, D, E에 저장하는 행렬 연산 함수 3개(addMtrx(), subtractMtrx(), multiplyMtrx())를 작성하라.

```
void addMtrx(double A[][SIZE], double B[][SIZE], double X[][SIZE], int size);
```

```
void subtractMtrx(double A[][SIZE], double B[][SIZE], double X[][SIZE], int size);
```

```
void multiplyMtrx(double A[][SIZE], double B[][SIZE], double X[][SIZE], int size);
```

- 2) 2차원 배열 A와 B를 각각 초기화하라. 초기화 데이터는 다음과 같이 설정할 것.

- 3) 초기화 된 2개의 행렬을 printMtrx(double M[][SIZE], int size)함수를 사용하여 출력할 것. 이 때, 행렬을 표시하기 위하여, 확장 완성형 코드를 사용할 것.

- 4) 행렬 A, B의 덧셈, 뺄셈, 곱셈을 위 1)에서 작성한 함수들을 사용하여 실행하라.

- 5) 행렬 연산을 총괄하는 main() 함수에서 2개의 행렬 초기화를 실행하고, printfMtrx(), addMtrx(), subtractMtrx(), multiplyMtrx() 함수를 호출하며, 행렬 준비, 행렬 연산 및 결과 출력을 실행하고, 그 결과를 확인할 것.

- 6) 실행 예제:

```
Matrix A :  
[ 5.00  4.00  3.00  2.00  1.00  
 6.00  7.00  8.00  9.00 10.00  
11.00 12.00 13.00 14.00 15.00  
16.00 17.00 18.00 19.00 20.00  
21.00 22.00 23.00 24.00 25.00 ]  
Matrix B :  
[ 1.00  0.00  0.00  0.00  0.00  
 0.00  1.00  0.00  0.00  0.00  
 0.00  0.00  1.00  0.00  0.00  
 0.00  0.00  0.00  1.00  0.00  
 0.00  0.00  0.00  0.00  1.00 ]  
Matrix C = A + B:  
[ 6.00  4.00  3.00  2.00  1.00  
 6.00  8.00  8.00  9.00 10.00  
11.00 12.00 14.00 14.00 15.00  
16.00 17.00 18.00 20.00 20.00  
21.00 22.00 23.00 24.00 26.00 ]  
Matrix D = A - B:  
[ 4.00  4.00  3.00  2.00  1.00  
 6.00  6.00  8.00  9.00 10.00  
11.00 12.00 12.00 14.00 15.00  
16.00 17.00 18.00 18.00 20.00  
21.00 22.00 23.00 24.00 24.00 ]  
Matrix E = A * B:  
[ 5.00  4.00  3.00  2.00  1.00  
 6.00  7.00  8.00  9.00 10.00  
11.00 12.00 13.00 14.00 15.00  
16.00 17.00 18.00 19.00 20.00  
21.00 22.00 23.00 24.00 25.00 ]
```

## <Oral Test>

Q3.1 enum을 사용하는 방법에 대하여 예를 들어 설명하라. enum으로 선언되는 기호 상수를 사용하여 요일 (weekday)의 이름과 달 (month)의 이름을 출력하는 방법에 대하여 설명하라.

(Key points: 요일과 달의 영어 이름을 각각 enum으로 열거)

Q3.2 반복문의 실행에서 continue와 break가 실행되면 어떤 결과가 나타나는지에 대하여 설명하라.

(Key points: for-loop내부에 continue와 break가 포함된 예를 구분하여 설명)

Q3.3 함수호출에서 2차원 배열 2개 (A, B)를 인수로 전달하고, 그 배열들에 대한 지정된 연산 (+, -, \*)을 수행하여 또 다른 2차원 배열 X로 반환하기 위한 함수 matrixOperation() 구현 방법에 대하여 설명하라.

(Key points: 이 함수 호출에 사용되는 인수 (argument/parameter) 구성과 함수 본문에서의 행렬 연산을 위한 다중 for-loop 구성 방법을 상세하게 설명할 것.)

Q3.4  $N \times K$  행렬 A와  $K \times M$  행렬 B의 곱셈 계산을 하여  $N \times M$  행렬 C에 저장하는 위한 반복문의 구성에 대하여 설명하고, 각 항목의 계산이 어떤 순서로 실행되는가에 대하여 그림으로 표현하고, 이를 설명하라.

(Key points: 3중 for-loop의 실행 단계별로 계산에 해당되는 배열 원소를 표시할 것)