



Programming Assignment #1

과목명	C 프로그래밍 (0441)
학과	컴퓨터공학부
학번	202211342
이름	이율원
제출일	2022/04/11

<문제 1> 맥주 소비 (beer.c)

문제 정의

하루 평균 맥주 소비 캔 수와 맥주 가격을 입력 받고, 한 해동안 소비하는 맥주의 총 캔 수, 총 칼로리량, 총 가격과 맥주로 인한 몸무게 증가량을 구하는 프로그램을 작성하자.

주요 변수 및 상수 설명

KG_PER_YEAR	1 년동안 매일 맥주 350ml 1 캔 섭취 시 증가하는 몸무게
YEAR	1 년의 일 수
BEER_KCAL	맥주 350ml 한 캔 당 칼로리
daily	1 일 평균 맥주 소비 캔 수를 입력받아 저장하는 double 형 변수
price	맥주 한 캔 평균 가격을 입력받아 저장하는 int 형 변수
total	계산된 총 맥주 소비 캔 수를 저장하는 double 형 변수

Idea 또는 알고리즘

1 년동안의 총 맥주 소비 캔 수(=total)는 (1 일 평균 소비 맥주 캔 수 = daily) * (365 일)을 통해 구하였고, 총 맥주의 칼로리는 (총 맥주 소비 캔 수 = total) * (150kcal)을 통해 구하였다. 맥주의 총 가격은 (총 맥주 소비 캔 수 = total) * (맥주 한 캔 평균 가격 = price)를 통해 구하였고, 맥주로 인한 몸무게 증가량은 (1 일 평균 맥주 소비 캔 수 = daily) * (6.80389kg)로 구할 수 있었다. 이렇게 구한 값을 출력형식에 맞게 출력하였다.

수행 결과

실행 결과 #1 1일 평균 맥주 소비 캔 수: 2.5 평균 맥주 가격: 1250 총 맥주 소비 캔: 912.50 총 맥주 칼로리량: 136875.00cals 몸무게 증가량: 17.01kg 총 맥주 소비 지출 액: 1140625원	실행 결과 #2 1일 평균 맥주 소비 캔 수: 4.25 평균 맥주 가격: 2500 총 맥주 소비 캔: 1551.25 총 맥주 칼로리량: 232687.50cals 몸무게 증가량: 28.92kg 총 맥주 소비 지출 액: 3878125원
실행 결과 #3 1일 평균 맥주 소비 캔 수: 3.75 평균 맥주 가격: 2750 총 맥주 소비 캔: 1368.75 총 맥주 칼로리량: 205312.50cals 몸무게 증가량: 25.51kg 총 맥주 소비 지출 액: 3764062원	실행 결과 #4 1일 평균 맥주 소비 캔 수: 10 평균 맥주 가격: 4000 총 맥주 소비 캔: 3650.00 총 맥주 칼로리량: 547500.00cals 몸무게 증가량: 68.04kg 총 맥주 소비 지출 액: 14600000원

실행 결과 #5
1일 평균 맥주 소비 캔 수: 7.5 평균 맥주 가격: 2750 총 맥주 소비 캔: 2737.50 총 맥주 칼로리량: 410625.00cals 몸무게 증가량: 51.03kg 총 맥주 소비 지출 액: 7528125원

실행 결과 #6
1일 평균 맥주 소비 캔 수: 4.8 평균 맥주 가격: 3200 총 맥주 소비 캔: 1752.00 총 맥주 칼로리량: 262800.00cals 몸무게 증가량: 32.66kg 총 맥주 소비 지출 액: 5606400원

Sample Run #1 의 데이터 셋을 비롯하여, 여러 샘플 데이터 셋을 입력해보았다. 주어진 출력 형식에 맞게 반올림되어 출력되는 것을 확인할 수 있었다.

토의 사항

`printf` 로 실수를 출력할 때, 소수점 아래의 숫자 개수를 정해주면 따로 반올림에 관한 처리를 하지 않아도 알아서 반올림되어 출력되는 것을 알 수 있었다.

또한 실수형을 소수점 없이 정수로 출력할 때, `%0.01f` 처럼 소수점 자리수를 0으로 설정하는 것이 아닌 `%d` 로 출력하려하면 올바른 값이 출력된다는 것도 알 수 있었다.

<문제 2> 윤년을 가정한 경우 (leap.c)

문제 정의

하루 평균 맥주 소비 캔 수와 맥주 가격을 입력 받고, 한 해동안 소비하는 맥주의 총 캔 수, 총 칼로리량, 총 가격과 맥주로 인한 몸무게 증가량을 구하는 프로그램을 작성하되, 윤년일 때를 고려해보자.

주요 변수 설명

KG_PER_DAY	매일 맥주 350ml 1 캔 섭취 시 증가하는 몸무게
YEAR	1 년의 일 수
BEER_KCAL	맥주 350ml 한 캔 당 칼로리
daily	1 일 평균 맥주 소비 캔 수를 입력받아 저장하는 double 형 변수
price	맥주 한 캔 평균 가격을 입력받아 저장하는 int 형 변수
total	계산된 총 맥주 소비 캔 수를 저장하는 double 형 변수
isLeapYear	윤년 여부 (윤년 → 1, 평년 → 0)를 입력받아 저장하는 int 형 변수

Idea 또는 알고리즘

처음 프로그램을 작성할 때에, 어떻게 해야 조건문을 사용하지 않고 윤년일 경우와 평년일 경우를 나누어 생각할 수 있을까 고민을 했었는데, 윤년이 평년보다 하루가 많고, 윤년일 경우 1 이, 평년일 경우 0 이 입력된다는 점에 착안하여 365 일에 isLeapYear 의 값을 더하는 것으로 조건문 없이도 윤년의 경우를 고려한 프로그램을 작성할 수 있었다. 1 년동안의 총 맥주 소비 캔 수(=total)는 (1 일 평균 소비 맥주 캔 수 = daily) * (365 일 + isLeapYear)을 통해 구하였고, 총 맥주의 칼로리는 (총 맥주 소비 캔 수 = total) * (150kcal)을 통해 구하였다. 맥주의 총 가격은 (총 맥주 소비 캔 수 = total) * (맥주 한 캔 평균 가격 = price)를 통해 구하였고, 맥주로 인한 몸무게 증가량은 (1 일 평균 맥주 소비 캔 수 = daily) * (6.80389kg / 365 일) * (365 일 + isLeapYear)로 구할 수 있었다. 이렇게 구한 값을 출력형식에 맞게 출력하였다.

수행 결과

실행 결과 #1 1일 평균 맥주 소비 캔 수: 2.5 평균 맥주 가격: 1250 윤년인지 아닌지 선택하시오(윤년=1, 평년=0): 0 총 맥주 소비 캔: 912.50 총 맥주 칼로리량: 136875.00cals 몸무게 증가량: 17.01kg 총 맥주 소비 지출 액: 1140625원	실행 결과 #2 1일 평균 맥주 소비 캔 수: 2.5 평균 맥주 가격: 1250 윤년인지 아닌지 선택하시오(윤년=1, 평년=0): 1 총 맥주 소비 캔: 915.00 총 맥주 칼로리량: 137250.00cals 몸무게 증가량: 17.06kg 총 맥주 소비 지출 액: 1143750원
실행 결과 #3 1일 평균 맥주 소비 캔 수: 0.285 평균 맥주 가격: 500 윤년인지 아닌지 선택하시오(윤년=1, 평년=0): 0 총 맥주 소비 캔: 104.02 총 맥주 칼로리량: 15603.75cals 몸무게 증가량: 1.94kg 총 맥주 소비 지출 액: 52012원	실행 결과 #4 1일 평균 맥주 소비 캔 수: 0.285 평균 맥주 가격: 500 윤년인지 아닌지 선택하시오(윤년=1, 평년=0): 1 총 맥주 소비 캔: 104.31 총 맥주 칼로리량: 15646.50cals 몸무게 증가량: 1.94kg 총 맥주 소비 지출 액: 52155원

실행 결과 #5
1일 평균 맥주 소비 캔 수: 3.14 평균 맥주 가격: 1592 윤년인지 아닌지 선택하시오(윤년=1, 평년=0): 0 총 맥주 소비 캔: 1146.10 총 맥주 칼로리량: 171915.00cals 몸무게 증가량: 21.36kg 총 맥주 소비 지출 액: 1824591원

실행 결과 #6
1일 평균 맥주 소비 캔 수: 3.14 평균 맥주 가격: 1592 윤년인지 아닌지 선택하시오(윤년=1, 평년=0): 1 총 맥주 소비 캔: 1149.24 총 맥주 칼로리량: 172386.00cals 몸무게 증가량: 21.42kg 총 맥주 소비 지출 액: 1829590원

Sample Run #2 와 Sample Run #3 의 데이터셋을 포함하여, 동일한 1 일 평균 맥주 소비 캔 수와 평균 맥주 가격에 윤년 여부만 바꾼 데이터셋 6 종류를 실행해 본 결과, 윤년 여부에 따라 총 맥주 소비 캔, 칼로리량, 몸무게 증가량과 소비 지출 액이 변함을 확인할 수 있었다.

토의 사항

조건문을 사용하는 대신 간단한 연산을 사용하는 등의 여러가지 방법으로 동일한 문제를 해결할 수 있다는 사실을 깨달았다. 또한 define 으로 기호 상수를 정의하여 사용할 때, 상수값만 사용할 수 있는 것이 아니라 수식도 사용할 수 있다는 것을 알 수 있었다.

<문제 3> 운동을 하자 (exercise.c)

문제 정의

하루 평균 맥주 소비 캔 수와 평균 걷는 시간을 입력받고, 맥주로 인한 칼로리를 모두 소모하는데에 필요한 총 운동 시간과, 주어진 걷는 시간 이외에 추가로 필요한 운동 시간을 구하고, 맥주로 인한 칼로리를 모두 소모하기 위해 하루에 해야하는 운동 시간을 구하는 프로그램을 작성해보자.

주요 변수 설명

KCAL_PER_MIN	4km/h 로 걸을 때, 1 분당 소모되는 칼로리
BEER_KCAL	맥주 350ml 한 캔 당 칼로리
YEAR	1 년의 일 수
HOURL_IN_SEC	1 시간 = 3600 초
MIN_IN_SEC	1 분 = 60 초
daily	1 일 평균 맥주 소비 캔 수를 입력받아 저장하는 double 형 변수
time	1 일 평균 걷는 시간을 분 단위로 입력받아 저장하는 int 형 변수
totalTime	총 운동 시간을 계산해 저장하는 int 형 변수
leftTime	추가로 필요한 운동 시간을 계산해 저장하는 int 형 변수
avgTime	평균 운동 시간을 계산해 저장하는 int 형 변수

Idea 또는 알고리즘

총 운동 시간은 (1 일 평균 맥주 소비 캔 수) * (150 kcal) * (365 일) / (150.0 / 40.0)으로 구하고, 추가 운동 시간은 (총 운동 시간)에서 (1 일 평균 운동 시간 * 365 일)을 빼어 구하고, 필요한 1 일 평균 운동 시간은 (추가 운동 시간) / (365 일)과 (1 일 평균 걷는 시간)을 더한 값으로 구하였다. 이렇게 구한 값은편의를 위해 60 을 곱해 작은 시간 단위인 초 단위로 저장해둔 후, 필요한 단위에 따라 적절한 상수를 곱하여 단위를 변환하여 출력할 수 있도록 했다.

(n 시간 n 분 n 초)로 출력할 때, 초 단위의 시간에서 3600 을 나누어 시간을 구하고, 초 단위의 시간을 3600 으로 나눈 나머지를 60 으로 나누어 분을 구하고, 초 단위의 시간을 60 으로 나눈 나머지를 구해 초를 구할 수 있었다.

수행 결과

실행 결과 #1
1일 평균 맥주 소비 캔 수: 1.2 1일 평균 걷는 시간(분 단위): 0 맥주 칼로리를 소모하는데 필요한 총 운동 시간: 17520.00분 (292시간 0분 0초) 추가로 필요한 운동 시간: 17520.00분 (292시간 0분 0초) 1일 평균 운동 시간: 48.00분 (0시간 48분 0초)

실행 결과 #2

```
1일 평균 맥주 소비 캔 수: 2.5
1일 평균 걷는 시간(분 단위): 60
맥주 칼로리를 소모하는데 필요한 총 운동 시간: 36500.00분 (608시간 20분 0초)
추가로 필요한 운동 시간: 14600.00분 (243시간 20분 0초)
1일 평균 운동 시간: 100.00분 (1시간 40분 0초)
```

실행 결과 #3

```
1일 평균 맥주 소비 캔 수: 3.14
1일 평균 걷는 시간(분 단위): 15
맥주 칼로리를 소모하는데 필요한 총 운동 시간: 45844.00분 (764시간 4분 0초)
추가로 필요한 운동 시간: 40369.00분 (672시간 49분 0초)
1일 평균 운동 시간: 125.60분 (2시간 5분 36초)
```

실행 결과 #4

```
1일 평균 맥주 소비 캔 수: 0.001
1일 평균 걷는 시간(분 단위): 0
맥주 칼로리를 소모하는데 필요한 총 운동 시간: 14.60분 (0시간 14분 36초)
추가로 필요한 운동 시간: 14.60분 (0시간 14분 36초)
1일 평균 운동 시간: 0.03분 (0시간 0분 2초)
```

실행 결과 #5

```
1일 평균 맥주 소비 캔 수: 2.7
1일 평균 걷는 시간(분 단위): 300
맥주 칼로리를 소모하는데 필요한 총 운동 시간: 39420.00분 (657시간 0분 0초)
추가로 필요한 운동 시간: -70080.00분 (-1168시간 0분 0초)
1일 평균 운동 시간: 108.00분 (1시간 48분 0초)
```

Sample Run #4 와 Sample Run #5 의 데이터 셋을 입력해 본 결과 출력값이 동일하게 나왔다. 운동 시간에서 초 단위의 계산도 잘 되는 것인지 궁금해 실행 결과 #4 의 데이터셋으로 확인해보았고, 맥주로 인한 칼로리보다 운동을 더 많이 하면 어떤 결과가 나올지 궁금해 실행 결과 #5 의 데이터 셋으로 테스트해보았다. 실행 결과 #5 에서는 의도하지 않은 결과가 나오는 것을 확인할 수 있었는데, 이는 추후 배운 조건문으로 예외 처리를 하여 해결할 수 있을 것이다.

토의 사항

기호 상수 `KCAL_PER_MIN` 을 정의할 때, $(150.0 / 40.0)$ 이 아닌 $150.0 / 40.0$ 으로 정의하게 되면 정확한 값이 나오지 않게 되는데, 이는 `define` 의 작동 원리 때문이다. 만약 `KCAL_PER_MIN` 을 `const double` 로 선언했다면, `KCAL_PER_MIN` 에 $150.0 / 40.0$ 의 연산값이 저장되어 사용되지만, `define` 으로 정의되었다면 컴파일 과정에서 소스코드의 모든 `KCAL_PER_MIN` 대신에 $150.0 / 40.0$ 이 치환되게 되고, `totalTime` 을 계산할 때 $(daily * 150 * 365) / (150.0 / 40.0)$ 이 아닌 $(daily * 150 * 365) / 150.0 / 40.0$ 을 계산한 것이 되어, 연산자 우선순위에 의해 원하는 연산값이 나오지 않게 된다.