

프로그래밍 과제 3

20222663 권기영

1. 단어 정렬

<구현 아이디어>

- 1) 입력은 fgets로 한줄씩 받고, end를 입력할 때까지 계속해서 입력을 받는다.
- 2) 입력한 한 줄에 대해서 strtok 함수를 사용해 공백문자를 기준으로 토큰화를 하고, 단어를 저장하는 배열에 토큰이 있는지 없는지를 직접 만든 find 함수를 통해 판단한 뒤, 배열에 토큰이 없다면 저장한다.
- 2) 정렬은 qsort로 하고, compare 함수가 중요한데, strcmp에 (char *)형으로 타입캐스팅한 인자를 넣어 비교를 진행했고, 사전순으로 정렬하였다.

<실행 예>

```
PS C:\Users\kweonkiyeong\Desktop\1-2_C_Programming> gcc .\word_sorting.c
PS C:\Users\kweonkiyeong\Desktop\1-2_C_Programming> .\a.exe
> once when i was six years old i
> saw a magnificent picture
> end

2 12 11

a
i
magnificent
old
once
picture
saw
six
was
when
years
PS C:\Users\kweonkiyeong\Desktop\1-2_C_Programming>
```

end를 입력받을 때 까지 문장을 입력받고, end를 입력받으면 순서대로 문장 개수, 입력 단어 개수, 저장 단어 개수를 출력한다.

또한 사전순으로 저장한 단어를 정렬한 뒤 출력한다.

2. 타자 게임

<구현 아이디어>

- 1) 무작위의 문장을 만드는 함수 generateStr을 구현한다. 문장과 단어 배열을 인자로 받아서 글자

개수(4 ~ 10)를 랜덤으로 정하고, 글자 개수에 맞는 단어에 들어가는 알파벳(A ~ Z)을 랜덤으로 정한다.

2) 무작위로 만들어진 단어들을 strcat 함수를 활용해서 문장의 뒤에 계속 연결한다. 만약 현재 문장의 길이 + 새로 추가될 글자 개수 가 50을 넘는다면 break 한다.

3) 사용자가 문장에 맞게 타자를 칠 때, 타자 속도를 반환하고 틀린 부분을 출력해주는 함수 evalTyping을 구현한다. 랜덤 문장, 사용자가 입력한 문장, 타자 시간을 인자로 받는다.

4) 랜덤 문장과 사용자가 입력한 문장 둘 중 하나가 null을 만날 때까지 문자 하나하나 비교하면서 같으면 점수를 1점 올리고, 다르면 그 자리에 오류를 출력한다. 오류를 출력할 때 주의해야 할 점은, 입력한 문장이 길이가 더 짧거나 더 길면, 그 부분은 무엇을 쓰던 무조건 오류로 표시해야 한다. 마지막에는 점수를 분당 타자횟수로 변환하여 반환한다.

5) 총 5번의 시행을 거친 뒤, 최종 점수를 출력한다.

<실행 예>

```
PS C:\Users\kweonkiyeong\Desktop\1-2_C_Programming> gcc .\typing_game.c
PS C:\Users\kweonkiyeong\Desktop\1-2_C_Programming> .\a.exe
게임을 시작합니다.
(1/5) 다음을 입력하세요
= ICSGQMSUG POJQICHW TCUTRB TMNTQHTEBF XXTCCK
= ICSGQMSUG POJQICHW TCUTRB TMNTQHTEBF XXTCCK

592타/분
(2/5) 다음을 입력하세요
= UJLRITBU NQRVP JFNTN HOHQL BDEDRLK RRBEGL
= UJLRITBU NQRVP JFNTN HOHQL BDEDRLK abcdGJ
      ^^^^

307타/분
(3/5) 다음을 입력하세요
= TDCOO BFIZXLOAT JJIWIYUZFH YWXH IIZCYFF PTVNWUA
= TDCOO BFIZXLOAT JJIWIYUZFH YWXH IIZCYFF PTVNWUAasdf
      ^^^^^

192타/분
(4/5) 다음을 입력하세요
= GQZWQFJL QPPHZD TKCCUQCQKA QTZHR GCPQDKDZ
=
  ^^^^^^^^^ ^^^^^^ ^^^^^^^^^^^^^ ^^^^^ ^^^^^^^^^
43타/분
(5/5) 다음을 입력하세요
= JYSIWOD MXOWQT VBSNWYYRQ CXNRMAJX YAZOB MOKQVA
= JYSIWOD MXOWQT VBSNWYYRQ CXNRMAJX YAZOB MOKQVA
      ^^^^^

263타/분
당신의 점수는 1397점 입니다.
PS C:\Users\kweonkiyeong\Desktop\1-2_C_Programming> |
```

1번 시행은 모두 동일하게 쳐서 오류를 출력하지 않는다. 2번 시행은 다른 것들에 대해서 오류를 출력하고 있다. 3번 시행은 주어진 문장을 초과한 입력을 했기 때문에 오류를 출력한다. 4번 문장은 전부 공백을 입력하여 공백을 제외한 모든 곳에서 오류를 출력한다. 5번은 문장과 동일하게 쳤지만, 마지막에 공백 5번을 추가로 입력하여 오류를 출력하는 모습이다.

5번의 시행을 모두 합친 결과를 마지막에 점수로 출력한다.

3. 80자리 계산기

<구현 아이디어>

- 1) 더하기와 빼기 모두 사람이 계산하는 알고리즘을 통해 계산하도록 한다., 즉 가장 작은 숫자부터 하나씩 숫자끼리 비교하여 계산하고, 더해서 자리수가 올라갈 때는 carry를, 빼서 음수가 나올 때는 borrow를 하는 알고리즘으로 구현한다.
- 2) 중요한 점은 문자열을 저장할 때 역순으로 저장해야 한다. 더하기를 해서 자리수가 올라 간다거나, 빼기를 해서 음수가 나와 -를 붙여야 하는 상황에서, 그냥 순서대로 저장된 문자열은 전부 오른쪽으로 밀고 첫번째 자리에 삽입을 해야 하기 때문이다. 따라서 일단 역순으로 저장하고, 문자열의 맨 끝에서 작업들을 수행한 뒤 다시 역순으로 바꿔 출력하는 방식을 사용한다.
- 3) 숫자의 길이가 차이 날 때는 더 짧은 쪽의 값을 0으로 만들고 계산한다.
- 4) 더하기를 수행하는 Add 함수를 구현한다. 같은 자리의 숫자끼리 더하고, 그 값이 10 이상이면 carry가 발생한 것이므로 carry를 1로 만들고, 값 - 10을 한다. 그런 식으로 마지막까지 더한 뒤, 마지막에 carry가 1이라면 $9999 + 1$ 처럼 자리수가 하나 올라간 것이므로 이에 대한 처리도 해준다.
- 5) 뺄셈을 수행하는 Subtract 함수를 구현한다. 같은 자리의 숫자끼리 빼고, 그 값이 0보다 작으면, borrow를 진행하고, 값 + 10을 한다.
- 6) Add 함수보다 살짝 복잡한데, 그 이유는 다음과 같다 1 - 결과가 음수일 때는 큰 수에서 작은 수를 뺀 뒤 맨 앞에 -를 붙여야 되기 때문에 두 개의 피연산자 중 어떤 수가 더 큰지를 판단하는 함수 compareStr을 직접 만들어야 한다. 2 - 예를 들어 $9999 - 9998$ 을 하면 0001(실제로는 이의 역순으로 저장됨)처럼 0이 앞에 붙은 채로 저장되기 때문에, 쓸모없는 0을 모두 날리는 작업을 추가로 진행해줘야 한다.
- 7) 입력을 받을 때, fgets로 81개를 받는데, 입력을 받은 뒤 rewind로 입력 버퍼를 초기화 시켜야 한다. 그 이유는, 사용자가 81개를 넘어서 입력을 하게 되면 fgets는 81개에서 끊고, 나머지가 입력 버퍼에 남아있는 상태로 다음 fgets에 들어가서 오류가 발생하기 때문이다.

<출력 예>

```

PS C:\Users\kweonkiyeong\Desktop\1-2_C_Programming> gcc .\bignum_calculation.c
PS C:\Users\kweonkiyeong\Desktop\1-2_C_Programming> .\a.exe
input X. 124513453245
input Y. 121524523434

X + Y = 246037976679
X - Y = 2988929811

input X. 131412412
input Y. 96374573457

X + Y = 96505985869
X - Y = -96243161045

input X. 1
input Y. 9999999999999999

X + Y = 10000000000000000
X - Y = -9999999999999998

input X.
input Y.
PS C:\Users\kweonkiyeong\Desktop\1-2_C_Programming>

```

```

PS C:\Users\kweonkiyeong\Desktop\1-2_C_Programming> python
Python 3.11.6 (tags/v3.11.6:8b6ee5b, Oct 2 2023, 14:57:12) [MSC v.1935 64 bit (AMD64)] on win32
Type "help", "copyright", "credits" or "license" for more information.
>>> 124513453245 + 121524523434
246037976679
>>> 124513453245 - 121524523434
2988929811
>>> 131412412 + 96374573457
96505985869
>>> 131412412 - 96374573457
-96243161045
>>> 1 + 9999999999999999
10000000000000000
>>> 1 - 9999999999999999
-9999999999999998
>>>

```

python에서 계산한 결과와 동일함을 확인할 수 있었다.