

윈도우 프로그래밍 워밍업 1

2024년도 1학기

1. [1차원 배열 사용하기] 문자열 다루기

- 영어 대소문자와 띄어쓰기만으로 이루어지고 마지막에 마침표로 마치는 문장을 입력받아 1차원 배열에 저장한다.
 - 문장의 길이: 최대 60자 (최대 길이는 변경 가능).
 - 단어는 띄어쓰기로 구분하는데 1개 이상의 연속 공백이 있는 경우는 1개의 공백으로 처리한다.
- 다음의 명령어를 실행한다. 명령어를 다시 입력하면 원래로 돌아간다.
 - a ~ z: 사용자가 입력하는 문자를 모두 * 로 바꾸고 출력
 - 1: 모든 공백에 다음 네 문자(@, %, # 또는 .) 중 한 개를 넣고 출력
 - 2: 단어의 개수를 출력 (공백을 기준으로 단어 분리한다)
 - 3: 대문자는 소문자로, 소문자는 대문자로 바꾸기
 - 4: 문장의 문자를 좌측으로 한 문자씩 이동한다.
 - 5: 문장의 문자를 우측으로 한 문자씩 이동한다.
 - 0: 프로그램을 종료

** 마침표는 항상 문장의 마지막에 놓여진다.

1. [1차원 배열 사용하기] 문자열 다루기

- 결과 예

Input sentence: C language is Base OF windoW prograMMing.

명령어 w: C language is Base OF *indo* prograMMing.

//--- w 를 * 로 바꾸기

명령어 w: C language is Base OF windoW prograMMing.

//--- 제자리로

명령어 e: C languag* is Bas* OF windoW prograMMing. //--- e 를 * 로 바꾸기

명령어 e: C language is Base OF windoW prograMMing.

//--- 제자리로

명령어 1: C@language%is@Base#of.windoW@prograMMing.

//--- 공백에 랜덤한 문자 삽입하기

명령어 2: Result: 7 words

//--- 단어의 개수 출력

명령어 1: C language is Base OF windoW prograMMing.

//--- 제자리로

명령어 3: c LANGUAGE IS bASE of WINDOw PROGRmmING.

//--- 대문자→ 소문자로, 소문자→ 대문자

명령어 3: C language is Base OF windoW prograMMing.

//--- 제자리로

명령어 4: language is Base OF windoW prograMMingC.

//--- 한 칸씩 왼쪽으로

명령어 4: language is Base OF windoW programmingC .

//--- 한 칸씩 왼쪽으로

명령어 4: anguage is Base OF windoW programming I .

//--- 한 칸씩 왼쪽으로

명령어 5: language is Base OF windoW programmingC .

//--- 한 칸씩 오른쪽으로

2. [1차원 배열 사용하기] 숫자 다루기

- 숫자를 입력하고 정렬, 검색 또는 다른 명령어를 수행한다.
 - 사용자가 입력하는 숫자보다 작은 랜덤 한 수를 50개 자동으로 만든다.
 - 사용자는 100보다 큰 수를 입력하고, 그 외에 숫자를 입력하면 에러메시지를 출력한다.
 - 만들어진 숫자는 중복 불가.
- 다음의 명령어를 실행한다.
 - a: 오름차순으로 정렬
 - d: 내림차순으로 정렬
 - e: 3의 배수 출력하고 그 숫자를 모두 (3의 배수) 더한 값도 출력한다.
 - f: 5의 배수 출력하고 그 숫자를 모두 (5의 배수) 더한 값도 출력한다.
 - m: 3의 배수 이거나 5의 배수인 숫자를 출력하고 그 숫자에 1을 더한 숫자도 출력한다.
 - n: 3의 배수 이고 5의 배수인 숫자를 출력하고 그 숫자에 1을 뺀 숫자도 출력한다.
 - r: 다시 재정렬
 - p: 다시 숫자를 입력 받고 기존과 같이 (입력 받은 값과 최대 값 사이) 50개 숫자를 랜덤하게 만들고 출력하기 (수정 부분)
 - Q: 프로그램 종료

*** 수정한 부분 대신 기존의 방법 (입력 수보다 큰 수로 랜덤하게 만들기)로 구현해도 무관함**

2. [1차원 배열 사용하기] 숫자 다루기

- 결과 예

Input number: 50

Generated numbers: 5 3 9 11 7 13 19 21 15 17 31 33 41 43 39 37 29 27 25 35

Command: a 3 5 7 9 11 13 15 17 19 21 25 27 29 31 33 35 37 39 41 43

Command: d 43 41 39 37 35 33 31 29 27 25 21 19 17 15 13 11 9 7 5 3

Command: r 5 3 9 11 7 13 19 21 15 17 31 33 41 43 39 37 29 27 25 35

Command: e 3 9 15 21 27 33 39 → 147 //--- 3의 배수, 147은 7개의 3의 배수를 모두 합한 값

Command: f 5 15 25 35 → 80 //--- 5의 배수, 80은 4개의 5의 배수를 모두 합한 값

Command: m 3 5 9 15 21 25 27 33 35 39 → 4 6 10 16 22 26 28 34 36 40 //--- 3의 배수 이거나 5의 배수인 숫자
//--- 1을 더한 값도 출력

Command: n 15 → 14 //--- 3의 배수이고 5의 배수인 수와 1을 뺀 수도 출력

Command: q 프로그램 종료