

C프로그램 기말고사 (2020-06-24)

1. 아래 왼쪽은 집합 A, B에 대한 차집합을 보여준다. 이를 아래 오른쪽과 같이 프로그램으로 출력하시오. 출력 시 순서는 상관없으나, 아래의 출력과 같이 표현 되어야 한다.
A, B, C(차집합 저장공간) 모두 int 배열로 선언하여 구현한다.

A = { 2, 11, 5, 7, 9, 15, 1 } B = { 13, 1, 5, 7, 13, 11, 3 } A - B = { 2, 9, 15 }	* 출력 예 A - B = { 2, 9, 15 }
---	--------------------------------

2. 사용자로부터 문자열을 입력 받은 후에, 문자열 중에서 양의 정수를 숫자로 변환하여 합계와 평균을 구한다. 숫자문자는 한 문자씩 양의 정수로 변환하고 연속된 숫자문자에 대해서도 한 문자씩 숫자로 인식한다. 또한 '0'도 숫자문자로 포함된다.

문자열 입력(1~99글자이내) : a012b456c78	* 출력 예 sum : 33 avg : 4.125
--------------------------------	-----------------------------------

3. A와 B 두 개의 정수 배열을 Bubble Sort로 각각 정렬을 하고, 이를 C 배열에 Merge Sort로 정렬하려고 한다. 중복되는 것은 하나만 C배열에 입력한다. (Bubble Sort : 강의자료실 버블소트리뷰 참고)
다음과 같이 제시된 main 함수와 함수원형을 이용해서 완성하시오.
A와 B 배열의 수는 5로 정하고 C는 10으로 정해서 사용한다. 별도로 size를 구하지 않는다.

<pre>void BubbleSort(int ary[], int len) { ... } int MergeSort(int arrA[], int arrB[], int arrC[]) { ... } int main(void) { int A[5] = { 11, 5, 2, 7, 1 }; int B[5] = { 3, 11, 13, 5, 9 }; int C[10] = { 0 }; BubbleSort(A, 5); BubbleSort(B, 5); int count = MergeSort(A, B, C); // count : MergeSort 중복 제외 원소수 // 출력부분 return 0; }</pre>	* 출력 예 C : 1 2 3 5 7 9 11 13
--	-------------------------------------

4. int 배열을 통해 Stack을 구현하고자 합니다. push(n)를 하면 한 개의 양의 정수가 int 배열에 들어가고, pop()을 하면 최근에 int 배열에 입력된 정수 한 개를 보내준다. int arr[10]로 고정했으므로 만일 push(n)를 했는데 이미 10개가 다 입력되었으면 "10번째 stack full"이라고 하고 0(false)을 반환한다. 그리고 만일 한 개도 입력된 것이 없을 때, pop()을 하면 "0번째 stack empty"라고 하고 0(false)을 반환한다. push(n)와 pop()이 성공하면 1(true)을 반환한다. 제시된 main() 함수가 보기와 같이 결과를 보여줄 수 있도록 push()와 pop()을 완성하시오.

<pre>#include <stdio.h> int tos = 0; // The Top of Stack (0에서 시작) int* parr; // The pointer of Stack (0에서 시작) int push(int p) { ... } int pop() { ... } int main(void) { int arr[10]; parr = arr; for (int i = 1; i < 12; i++) { if(push(i)) printf("%d ", *(parr - 1)); // push() 값 확인 else printf("\n%d번째 stack full\n", i); // 실패 0 } for (int i = 10; i >= 0; i--) { if (pop()) printf("%d ", *parr); // pop() 값 확인 else printf("\n%d번째 stack empty\n", tos); } for (int i = 1; i <= 3; i++) { push(i*10); printf("%d ", *(parr-1)); // push() 값 확인 } return 0; }</pre>	<p>* 출력 예</p> <pre>1 2 3 4 5 6 7 8 9 10 11번째 stack full 10 9 8 7 6 5 4 3 2 1 0번째 stack empty 10 20 30</pre> <div style="text-align: center; margin-top: 20px;"> </div>
--	--

5. 아래와 같을 때 배열과 포인터의 같은 점과 다른 점을 설명하시오.

```
int arr[10] = { 0 };
int * parr;
parr = arr;
```