2016학년도 수행평가 1차시(포인터&정렬검색알고리즘)

학버	:	이름 :	

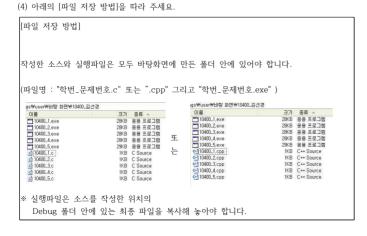
- (1) 컴퓨터 바탕화면에 띄어진 모든 프로그램을 종료하세요.
- (2) 바탕화면에 폴더(폴더명: "학번_이름")를 만드세요.

(급) 10400_김선경

(3) 아래 주의사항을 읽어주십시오.

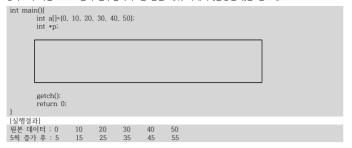
[주의사항]

- 1. Visual Studio 이외의 프로그램 (웹브라우저, 메신저 등)을 실행하고 있을 경우, 부정행위로 간주되어 0점 처리 됩니다.
- 다 푼 학생은 조용히 모니터를 끄고 대기합니다.
 NetClass 파일을 전송하기 위한 메세지는 수업 종료 5분 전에 보내드립니다.
- NetClass 파일 전송 시 바탕화면의 학번_이름 폴더를 압축하여 전송하세요. 파일이름: 학번_이름 파일명 : 학번_이름.zip
- 4. 실습 문제는 5문항(1~5번), 지필문제 5문항(6~10번)의 총 10문제가 주어집니다. 꼭 순서대로 풀 필요는 없습니다. 코딩 오류로 인하여 파일이 실행되지 않으면 0점 처리 됩니다.



是进1。

다음은 <u>포인터 p를 이용</u>하여 1차원 배열에 있는 모든 원소들을 5씩 증가시키는 프로그램을 작 성하고자 하는 프로그램의 일부입니다. 빈 칸을 채워 아래의 [실행결과]를 얻으세요.



是周2

5개의 데이터를 배열에 입력받아 배열의 내용을 내림차순 정렬하는 프로그램을 작성한 후 아 래의 [실행결과]를 얻으세요.

[실행결과]

```
배열의 내용 입력 :
22 44 66 33 77[Enter-]
```

문제3.

키보드로 임의의 문자열을 입력하면 이것을 거꾸로 출력하는 프로그램을 <u>포인터를 이용</u>하여 작성한 후 아래의 [실행결과]을 얻으세요.

[식행격과]

```
입력데이터 : ABCDEFGEnter-
```

5 페이지 중 1

문제4.

문자열 str에 포함되어 있는 숫자 문자 '0'~'9'의 개수를 cnt[0]~cnt[9]의 값과 인덱스를 일치시 켜 각각 저장하는 void cntDigit(const char *str, int cnt[])를 작성하여 아래의 [실행결과] 를 얻으세요.

```
- - ...
[실행결과]
문자열 입력 : web 1-4, 1-5, 1-6[Enter-]
숫자 문자의 사용횟수 출력
0의 사용횟수: 0 1의 사용횟수: 3 2의 사용횟수: 0 3의 사용횟수: 0 4의 사용횟수: 1
5의 사용횟수: 1 6의 사용횟수: 1 7의 사용횟수: 0 8의 사용횟수: 0 9의 사용횟수: 0
```

문제5.

1~25까지의 수 중 랜덤으로 발생된 10개의 숫자로 초기화 된 배열에서 사용자가 입력한 숫자 가 배열에서 몇 번째에 위치하고 있는지 이분검색(Binary Search)의 방법으로 찾는 프로그램 을 작성 후 [실행결과]를 얻으세요.(단, 정렬 알고리즘은 자유롭게 선택하여 작성한다.)

[실행결과]

```
배열 출력 :
 6 12 4 15 22 25 1 18 13 10
찾을 숫자 입력: 13Ember-1
 <<<오름차순 정렬을 먼저 수행합니다.
 1 4 6 10 12 13 15 18 22 25
BinarySearch(검색 횟수): ■■■
 13는 배열에서 6번째에 위치하고 있습니다.
```

>>>다음 면부터는 [지필평가]문제입니다. 실습 문제에 대한 파일 저장을 잘 확인하세요.

```
※ 문제 6~10번은 지필평가 문제입니다.
```

각 문제를 잘 읽고, 정답을 시험지에 직접 작성하세요.

문제6.

n=3이고, 배열 A는 [16, 30 ,2]와 같을 때 아래의 버블 정렬(Bubble Sort)을 이용하여 정렬하고 자합니다. 이 때 swap()함수는 몇 번 호출되는지 적어보세요.

```
for (j = 0; j < n-1; j++)
  for (i = 0; i < n-j-1; i++)
    if (A[i] > A[i+1])
    swap(A[i], A[i+1]);
```

문제7.

삽입 정렬(Insertion Sort)의 정렬 기법을 이용하여 다음의 정수 배열이 오름차순으로 정렬되는 모습을 나타내고자 합니다. 3단계의 출력 내용을 채워보세요.

```
데이터 : <mark>31 28 10 13 2 20 7 5</mark>
```

```
for (i = 1; i <= 3; i++) { // 3단계 설정
e = A[i]; j = i;
while (j > 0) {
if (A[j-1] > e)
A[j] = A[j-1];
else
break;
j--;
}
A[j] = e;
}
```

100 (1) (2) (3) (4) (5) (6) (7) 3단계:

5 페이지 중 3

문제8.

선택 정렬(Selection Sort)의 정렬 기법을 이용하여 다음의 정수 배열이 오름차순으로 정렬되는 모습을 나타내고자 합니다. 3단계의 출력 내용을 채워보세요.(n : 배열의 요소 수)

 (0)
 (1)
 (2)
 (3)
 (4)
 (5)

 3단계 :

문제9.

다음 6개의 정수를 퀵 정렬(Quick Sort)의 정렬 기법을 이용하여 오름차순 정렬하고자 하였습니다. <보기>의 배열이 퀵 정렬 알고리즘의 첫번째 단계를 끝나고 나서 기준값(pivot)에 의해 분할된 결과라고 가정하면, 기준값(pivot)으로 사용하였을 가능성을 가진 정수는 몇 개일까요?

(0) (1) (2) (3) (4) (5) 9 8 2 9 22 30	<보기>						
9 8 2 9 22 30		[0]	[1]	[2]	[3]	[4]	[5]
		9	8	2	9	22	30

(

是和10.

다음은 힙정렬(Heap Sort)를 위해 데이터를 힙구조로 만드는 알고리즘입니다. 이 알고리즘에 의해 힙(Heap)이 구성된 모습을 보이세요.

	[1]	[2]	[3]	[4]	[5]	[6]
데이터 :	26	5	37	1	61	11

[합 구조 만드는 알고리즘] 요소수가 N개일 때, 데이터 열(D₁, D₂, ..., D_n)에 대하여 1단계: 1=12/2 때, 어대의 3단계를 반복한다. 3단계: D₇, D₂₇, D₂₇,를 비교한다. D₇가 최대일 때: I=1-1, T=1로 한다. D₂₇가 최대일 때: D₄와 D₂₇를 교환하여, T=2T로 한다. D₂₇₊₁가 최대일 때: D₅와 D₂₇를 교환하여, T=2T+1로 한다. 자식D₂₇, 자식D₂₇₊₁가 없을 때: I=I-1, T=I로 한다.

[1] [2] [3] [4] [5] [6] 완성된 힙 :