

2016학년도 수행평가 1차시(포인터&정렬검색알고리즘)

학번 : _____ 이름 : _____

(1) 컴퓨터 바탕화면에 찍어진 모든 프로그램을 종료하세요.

(2) 바탕화면에 폴더(폴더명: "학번_이름")를 만드세요.



(3) 아래 주의사항을 읽어주십시오.

[주의사항]

1. Visual Studio 이외의 프로그램 (웹브라우저, 메신저 등)을 실행하고 있을 경우, 부정행위로 간주되어 0점 처리 됩니다.
2. 다 푼 학생은 조용히 모니터를 끄고 대기합니다.
NetClass 파일을 전송하기 위한 메시지는 수업 종료 5분 전에 보내드립니다.
3. 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.

(4) 아래의 [파일 저장 방법]을 따라 주세요.

[파일 저장 방법]

작성한 소스와 실행파일은 모두 바탕화면에 만든 폴더 안에 있어야 합니다.

(파일명 : "학번_문제번호.c" 또는 ".cpp" 그리고 "학번_문제번호.exe")

gsWuserW\바탕 화면\10400_김선경			gsWuserW\바탕 화면\10400_김선경		
이름	크기	종류	이름	크기	종류
10400.1.exe	29KB	응용 프로그램	10400.1.exe	29KB	응용 프로그램
10400.2.exe	29KB	응용 프로그램	10400.2.exe	29KB	응용 프로그램
10400.3.exe	29KB	응용 프로그램	10400.3.exe	29KB	응용 프로그램
10400.4.exe	29KB	응용 프로그램	10400.4.exe	29KB	응용 프로그램
10400.5.exe	29KB	응용 프로그램	10400.5.exe	29KB	응용 프로그램
10400.1.c	1KB	C Source	10400.1.cpp	1KB	C++ Source
10400.2.c	1KB	C Source	10400.2.cpp	1KB	C++ Source
10400.3.c	1KB	C Source	10400.3.cpp	1KB	C++ Source
10400.4.c	1KB	C Source	10400.4.cpp	1KB	C++ Source
10400.5.c	1KB	C Source	10400.5.cpp	1KB	C++ Source

* 실행파일은 소스를 작성한 위치의
Debug 폴더 안에 있는 최종 파일을 복사해 놓아야 합니다.

문제1.

다음은 포인터 p를 이용하여 1차원 배열에 있는 모든 원소들을 5씩 증가시키는 프로그램을 작성하고자 하는 프로그램의 일부입니다. 빈 칸을 채워 아래의 [실행결과]를 얻으세요.

```
int main(){
    int a[]={0, 10, 20, 30, 40, 50};
    int *p;

    

    getch();
    return 0;
}
```

[실행결과]

원본 데이터 : 0	10	20	30	40	50
5씩 증가 후 : 5	15	25	35	45	55

문제2.

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

[실행결과]

배열의 내용 입력 :
22 44 66 33 77

정렬 후 배열 출력 :
77 66 44 33 22

문제3.

키보드로 임의의 문자열을 입력하면 이것을 거꾸로 출력하는 프로그램을 포인터를 이용하여 작성한 후 아래의 [실행결과]을 얻으세요.

[실행결과]

입력데이터 : ABCDEFG

출력데이터 : GFEDCBA

문제4.

문자열 str에 포함되어 있는 숫자 문자 '0'~'9'의 개수를 cnt[0]~cnt[9]의 값과 인덱스를 일치시켜 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

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 숫자가 배열에서 몇 번째에 위치하고 있는지 이분검색(Binary Search)의 방법으로 찾는 프로그램을 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 : 13

<<<오름차순 정렬을 먼저 수행합니다.

1 4 6 10 12 13 15 18 22 25

BinarySearch(검색 횟수) : ■■■■

13는 배열에서 6번째에 위치하고 있습니다.

문제10.

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
----	---	----	---	----	----

[힙 구조 만드는 알고리즘]

요소수가 N개일 때, 데이터 열(D_1, D_2, \dots, D_n)에 대하여

1단계 : $I=N/2, T=I$ 로 한다.

2단계 : $I \geq 1$ 일 때, 아래의 3단계를 반복한다.

3단계 : D_I, D_{2I}, D_{2I+1} 를 비교한다.

D_I 가 최대일 때 : $I=I-1, T=I$ 로 한다.

D_{2I} 가 최대일 때 : D_I 와 D_{2I} 를 교환하여,
 $T=2I$ 로 한다.

D_{2I+1} 가 최대일 때 : D_I 와 D_{2I+1} 를 교환하여,
 $T=2I+1$ 로 한다.

자식 D_{2I} , 자식 D_{2I+1} 가 없을 때 : $I=I-1, T=I$ 로 한다.

[1]

[2]

[3]

[4]

[5]

[6]

완성된 힙 :

--	--	--	--	--	--