**1문제)**

텍스트, 만화 영화, 스크린샷, 클립아트이(가) 표시된 사진

자동 생성된 설명

메모리는 바이트(Byte)를 기준으로 주소가 매겨진다.

포인터는 변수의 주소 값을 저장할 수 있는 변수이다.

변수 x의 주소를 추출하려면 &x라고 하면 된다.

\*p의 의미는 p가 가리키는 값을 가져오라는 것이다.

int형 포인터 p가 가리키는 위치에 100을 저장하는 문장은 int \*p = 100;이다.

포인터가 아무것도 가리키고 있지 않는 경우에는 NULL 값을 넣어두는 편이 좋다.

배열 a에서 a는 첫번째 요소의 주소이다.

p가 포인터라면 p[2]는 수식 \*(p+2)와 같다.

\*p++의 의미는 p가 가리키는 값을 사용한 후에 p를 증가하는 것이다.

사칙 연산 중에서 포인터에 대하여 적용할 수 있는 연산에는 +, -, \*, ++, --이 있다. *(‘사칙연산’이라 +, -만 써야 되는지 몰라서 증감연산자까지 다 적었다.)*

int형 포인터 p가 80번지를 가리키고 있었다면 (p+1)은 84번지를 가리킨다.

함수 호출 시 인수 전달 방법 중에서 기본적인 방법은 “값에 의한 호출(call by value)”이다.

**2문제)**

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

char code;

char \*p = NULL; // char형 포인터 p선언 (주소를 넣지 않을 때는 NULL 사용)

p = &code; // 포인터에 변수 code의 주소 대입

\*p = ‘a’; // 포인터를 통하여 변수 code에 ‘a’ 대입하기

**3문제)**

텍스트, 영수증, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

**답 ) 3번**

풀이) 1, 2, 4번 모두 i의 값이 1증가하고 3번의 경우 p가 가리키는 값인 i를 사용한 후에 p의 가리키는 주소 값을 1칸(4byte) 증가시키기 때문에 i의 값이 그대로 사용되어 답이 3번이 된다.

**4문제)**

텍스트, 폰트, 화이트, 스크린샷이(가) 표시된 사진

자동 생성된 설명

**답) 5**

**5**

풀이) p가 x의 주소를 가리키고 –-(\*p)를 할 경우 p의 주소의 값이 6이므로 --6과 같은 의미가 된다. 따라서 먼저 증감연산자 -- 가 실행되어 5가 출력된다. 그 뒤에 (\*p)++를 할 경우 5++가 되기 때문에 5의 값이 출력되고 그 뒤에 값이 6이 된다.

**5문제)**

**텍스트, 폰트, 화이트, 대수학이(가) 표시된 사진

자동 생성된 설명**

**답) 1008**

**2008**

풀이) int형 포인터는 한 칸에 4byte이기 때문에 1000 + 4 \* 2 = 1008이 되어 1008이 출력되고 double형 포인터는 한 칸에 8byte이기 때문에 2000 + 8 = 2008이 되어 2008이 출력된다.

**6문제)**

**텍스트, 스크린샷, 폰트, 라인이(가) 표시된 사진

자동 생성된 설명**

**문제분석)**

먼저 이 코드의 경우 Call by Value이기 때문에 함수 내에서는 x와 y의 값이 서로 바뀌지만 main절에서 a와 b의 값은 바뀌지 않을 것이다. 그리고 값을 바꿀 때도 파이썬에서는 간단하게 값을 서로 바꿀 수 있지만 C언어나 다른 언어에서는 대부분 임시변수를 이용하여 값을 서로 바꿔야 한다. 그러면 한번 변수들의 값을 디버깅을 통해 확인해보자.

**소스코드)**

텍스트, 스크린샷, 소프트웨어, 폰트이(가) 표시된 사진

자동 생성된 설명

18번째 줄에서 swap(a, b) 함수 선언

스크린샷, 텍스트이(가) 표시된 사진

자동 생성된 설명

함수에 값이 잘 넘어감

스크린샷, 텍스트이(가) 표시된 사진

자동 생성된 설명

tmp에 x값 대입

스크린샷, 텍스트이(가) 표시된 사진

자동 생성된 설명

x에 y값 대입

스크린샷, 텍스트이(가) 표시된 사진

자동 생성된 설명

y에 tmp값 대입(결론적으로 x와 y는 서로 바뀜)

스크린샷, 텍스트, 블랙이(가) 표시된 사진

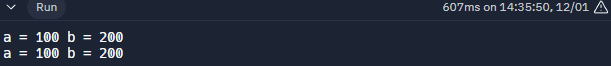
자동 생성된 설명

하지만 main에서의 a와 b는 바뀌지 않음(Call by Value)

스크린샷, 텍스트, 블랙이(가) 표시된 사진

자동 생성된 설명

**실행결과)**



**7문제)**

텍스트, 스크린샷, 라인, 폰트이(가) 표시된 사진

자동 생성된 설명

**문제분석)**

이 코드는 Call by Reference이기 때문에 함수 내에서는 a의 주소를 가리키는 매개변수 px와 b의 주소를 가리키는 매개변수 py가 함수 내에서 서로 바뀜에 따라 가리키는 값들이 서로 바뀌면서 main절에서의 a와 b도 서로 바뀔 것이다. 이것도 디버깅을 통해 확인해보자.

**소스코드)**

텍스트, 스크린샷, 폰트, 소프트웨어이(가) 표시된 사진

자동 생성된 설명

18번째 줄에서 swap(&a, &b) 함수 선언

스크린샷, 텍스트, 블랙이(가) 표시된 사진

자동 생성된 설명

함수에 주소 값이 잘 넘어감

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

tmp에 \*px의 값인 100이 잘 들어감

텍스트, 스크린샷, 폰트이(가) 표시된 사진

자동 생성된 설명

\*px에 \*py값 대입

텍스트, 스크린샷이(가) 표시된 사진

자동 생성된 설명

\*y에 tmp값 대입

텍스트, 스크린샷이(가) 표시된 사진

자동 생성된 설명

main절의 a, b값도 잘 바뀜(Call by Reference)

스크린샷, 텍스트, 블랙이(가) 표시된 사진

자동 생성된 설명

**실행결과)**

