

[부스트코스] 모두를 위한 컴퓨터 과학 첫 걸음 퀴즈 해설

3. 배열

Quiz 1

컴파일링의 네 단계에 해당하지 않는 것은 무엇인가요?

어셈블링(asmbling)

링킹(linking)

전처리(preprocessing)

디버깅(debugging)

-> 컴파일링의 네 단계는 다음과 같습니다. 전처리 -> 컴파일링 -> 어셈블링 -> 링킹. 디버깅은 프로그래밍을 하는 와중에 발생한 오류나 실수를 수정하는 과정을 의미합니다.

Quiz 2

소스코드 내에 존재하는 오류를 해결하기 위한 작업을 무엇이라고 하나요?

리졸빙(resolving)

디버깅(debugging)

컴파일링(compiling)

체킹(checking)

-> 디버깅이 프로그램에 내에 존재하는 오류나 실수를 해결하는 과정을 의미합니다. 리졸빙이나 체킹은 관습적으로 쓰이는 단어는 아닙니다. 컴파일링은 사람이 작성한 소스코드를 컴퓨터가 실행할 수 있는 머신코드로 바꿔주는 것을 의미합니다.

Quiz 3

아래 코드는 기능상으로는 문제가 없지만 C 언어 스타일 가이드에는 부합하지 않습니다. 어떻게 변경하면 스타일 가이드에 부합한 코드가 될까요? (코드 생략)

여는 중괄호({)와 같은 줄에 printf 함수를 작성한다.

int main(void) 바로 옆에서 여는 중괄호({)를 시작한다.

printf 함수와 세미콜론(;) 사이에 공백을 추가한다.

printf 함수를 들여쓰기해서 중괄호{} 사이에 포함되었음을 표현한다.

-> 문제의 코드의 기능상으로는 이상이 없지만 C 언어 스타일 가이드에 부합하려면 아래와 같이 수정해줘야합니다.

```
int main(void)
{
    printf("hello, world\n");
}
```

가독성 높은 코드를 작성하는 것은 프로그램 가독성 향상, 유지보수 그리고 협업에 굉장히 중요한 요소입니다.



프로그램을 시작하는 첫 단계부터 권장 스타일에 맞춰 코드를 작성하는 습관을 들이는 것이 좋습니다.

Quiz 4

char 자료형을 5개 담을 수 있는 [edwith]라는 이름의 배열을 선언하는 코드는 무엇인가요?

char edwith[4];

char edwith[5];

chars edwith[5];

char edwith*5;

-> 배열을 선언하는 방식은 "자료형 배열이름[배열크기];" 입니다. char 자료형을 가지므로 char, 배열 이름이 edwith이므로 char edwith, 배열의 크기가 5이므로 char edwith[5] 이 됩니다. 그리고 코드 마지막에는 세미콜론(;)을 붙여줍니다.

Quiz 5

`int scores[10];` 이라는 배열이 선언되어 있습니다. 이 배열의 처음부터 끝 인덱스까지 순환하는 for 루프는 무엇인가요?

`for (int i = 1; i < 10; i++) {... scores[i] ...}`

`for (int i = 0; i < 10; i++) {... scores[i] ...}`

`for (int i = 1; i < 11; i++) {... scores[i] ...}`

`for (int i = 0; i < 11; i++) {... scores[i] ...}`

-> 배열을 정의한 코드로부터 배열의 크기가 10이라는 것을 알 수 있습니다. for 루프를 사용해서 배열의 인덱스인 0부터 9까지 순환하는 코드를 작성해야 합니다. 따라서 i는 0부터 시작해서 1씩 증가시켜가면서 10보다 작아야 합니다.

Quiz 6

위와 같은 코드가 주어졌을 때 아래 보기 중 그 값이 다른 것은 무엇일까요? (코드 생략)

names[0][0]

names[1][0]

names[0][2]

names[1][2]

-> names라는 배열 안에 문자열이 저장되어 있습니다. 그리고 그 문자열은 문자로 이루어진 배열입니다. 따라서 names[0][0]은 "MONSTER"의 첫 번째, 즉 'M'이 됩니다. names[1][0]이나 names[1][2]도 마찬가지로 'M'입니다. 하지만 names[0][2]는 "MONSTER"의 세 번째, 즉 'N'이 됩니다.

Quiz 7

아래와 같은 main 함수를 지니는 edwith.c 코드소스를 작성하고 컴파일했습니다. '\$./edwith hello world'를 실행하였을 때 argc의 값은 무엇일까요? (코드 생략)

1

2

3

4

-> main 함수의 입력값으로 정의된 argc는 프로그램 실행시 전달되는 인자의 개수, argv는 인자를 배열로 저장합니다. 따라서 './edwith hello world'를 실행하면 프로그램명(edwith) , 입력값1(hello) , 입력값2(world) 세 개의 인자가 전달되고 argc에는 3이 저장됩니다. argv에 저장되는 인자는 띄어쓰기로 구분됩니다.

Quiz 8

C 소스파일의 링크, 컴파일, 빌드 작업을 자동화해주는 툴은 다음 중 무엇인가요?

clang

autocompile

make

link

-> make는 clang에서의 링크와 object 파일 생성과 같은 옵션을 지정해주지 않아도 자동으로 링크, 컴파일, 빌드 과정을 수행해 실행 프로그램을 생성해주는 툴입니다.

Quiz 9

소스코드를 작성할 때 우리가 의식적으로 또는 무의식적으로 저지른 실수나 오류를 통상적으로 일컫는 단어는 무엇인가요?

미스테인크(mistake)

버그(bug)

앱노멀(abnormal)

폴트(fault)

-> 초창기 컴퓨터에는 빛과 열을 보고 몰려드는 벌레(bug)로 인한 작동 오류가 많았고, 이때부터 프로그래밍에 발생한 오류를 '버그'라는 단어로 부르게 됩니다. 또 이러한 실수나 오류를 바로잡는 것을 '디버깅(debugging)'이라고 합니다.

Quiz 10

다음 중 문자열(string)이 메모리에 저장되는 방식에 대한 설명으로 틀린 것은 무엇인가요?

문자열(string)은 문자(char)의 배열로 저장된다

문자열(string)의 각 문자(char)는 메모리상에 연이어서 저장된다

문자열(string)은 포함된 문자(char) 개수 만큼의 메모리 자리를 차지한다

문자열(string)의 끝에는 널(NUL) 종단 문자가 포함된다

-> 문자열은 포함된 문자의 수에 더해서, 널(NUL) 종단 문자(\0)이 포함됩니다. 따라서 포함된 문자 수 + 1만큼의 메모리 자리를 차지합니다.