Question 1 (1 point)

Consider the following statement:

```
int* my_var;
```

this is an example of...

○ a declaration of an integer pointer variable

○ an expression that evaluates to the address of a variable

○ none of the above

○ an expression that dereferences a pointer variable

Question 2 (1 point)

Consider the following expression:

```
*my_var
```

this expression is an example of...

○ a declaration of an integer pointer variable

○ an expression that dereferences a pointer variable

○ an expression that evaluates to the address of a variable

○ none of the above

Question 3 (1 point)

Consider the following expression:

```
&my_var
```

this expression is an example of...

○ an expression that dereferences a pointer variable

○ a declaration of an integer pointer variable

○ none of the above

○ an expression that evaluates to the address of a variable

Question 4 (1 point)

Consider the following variable declarations:

```
int x;
int* y;
```

Assume the code to initialize these values exists but is omitted for this question.
What type is `x`

○ `int`

○ `int*`

Question 5 (1 point)

Consider the following variable declarations:

```
int x;
int* y;
```

Assume the code to initialize these values exists but is omitted for this question.
What type is `y`

○ `int`

○ `int*`

Question 6 (1 point)

Consider the following variable declarations:

```
int x;
int* y;
```

Assume the code to initialize these values exists but is omitted for this question.
What type is `&x`

○ `int`

○ `int*`

Question 7 (1 point)

Consider the following variable declarations:

```
int x;
int* y;
```

Assume the code to initialize these values exists but is omitted for this question.
What type is `*y`

○ `int`

○ `int*`

Question 8 (1 point)

What is the output of the following code snippet
NOTE: be sure to include the space between the values printed

```
int a = 2;
int* b = &a;
*b = 7;
printf("%d %d", a, *b);
```

Question 9 (1 point)

The following code snippet is the same as that in the last question, with the bolded line added. What is the output?
NOTE: be sure to include the space between the values printed

```
int a = 2;
int* b = &a;
*b = 7;
a++;
printf("%d %d", a, *b);
```

Question 10 (1 point)

The following code snippet is the same as that in the last question, with the bolded line added. What is the output?
NOTE: be sure to include the space between the values printed

```
int a = 2;
int* b = &a;
*b = 7;
a++;
int c = *b;
printf("%d %d %d", a, *b, c);
```

Question 11 (1 point)

The following code snippet is the same as that in the last question, with the bolded line added. What is the output?
NOTE: be sure to include the space between the values printed

```
int a = 2;
int* b = &a;
*b = 7;
a++;
int c = *b;
b = &c;
a++;
printf("%d %d %d", a, *b, c);
```

Question 12 (1 point)

The following code snippet is the same as that in the last question, with the bolded line added. What is the output?
NOTE: be sure to include the space between the values printed

```
int a = 2;
int* b = &a;
*b = 7;
a++;
int c = *b;
b = &c;
a++;
c++;
printf("%d %d %d", a, *b, c);
```

Question 13 (1 point)

The following code snippet is the same as that in the last question, with the bolded line added. What is the output?
NOTE: be sure to include the space between the values printed

```
int a = 2;
int* b = &a;
*b = 7;
a++;
int c = *b;
b = &c;
a++;
*b = a + *b;
printf("%d %d %d", a, *b, c);
```

Question 14 (1 point)

Given the following definition of the function `foo` (prototype and documentation omitted intentionally):

```
void foo(int a, int* b) {
    a = 20;
    *b = 50;
}
```

What is the output if the following code snippet is run:
NOTE: be sure to include the space between the values printed

```
int x = 1;
int y = 2;
foo(x, &y);
printf("%d %d", x, y);
```

Submit Quiz *0 of 14 questions saved*