Q-18

Due No due date **Points** 15 **Questions** 15 **Time Limit** 30 Minutes **Allowed Attempts** Unlimited

Take the Quiz Again

Attempt History

	Attempt	Time	Score
KEPT	Attempt 2	16 minutes	15 out of 15
LATEST	Attempt 3	12 minutes	14 out of 15
	Attempt 2	16 minutes	15 out of 15
	Attempt 1	21 minutes	14 out of 15

ⓘ Correct answers are hidden.

Submitted Jul 22 at 11:08pm

Question 1

1 / 1 pts

All of these are legal C++ statements; which of them uses the C++ ***dereferencing operator***?

```
int a = 3, b = 4;
```

- None of these use the dereferencing operator.
- `int y = a * b;`
- `z *= a;`
- `int x = *p;` ✓
- `int *p = &b;`

Question 2

1 / 1 pts

What is true about this code?

```
int n{500};
int *p = &n;
```

- *p is the value of n ✓
- p stores the same value as n
- &p is the direct or explicit value of n
- &n is the indirect value of p
- &p represents the indirect value of n

Question 3

1 / 1 pts

What is true about this code?

```
int * choice;
```

- `choice` contains an undefined address ✓
- Syntax error; should be `int choice*;`
- `choice` can point to any kind of object
- `choice` currently points to an integer
- `choice` currently contains an integer

Question 4

1 / 1 pts

Assume that ***ppi*** correctly points to ***pi***. Which line prints the ***size*** (in bytes) of ***pi***?

```
int main()
{
    double pi = 3.14159;
    double *ppi;
    // code goes here
    // code goes here
}
```

- `cout << sizeof(ppi);`
- None of these
- `cout << sizeof(&ppi);`

- cout << sizeof(*ppi);
- cout << sizeof(*pi);

Incorrect

Question 5

0 / 1 pts

Assume that ***ppi*** correctly points to ***pi***. Which line prints the value stored inside ***pi***?

```
int main()
{
    double pi = 3.14159;
    double *ppi;
    // code goes here
    // code goes here
}
```

- None of these
- cout << &ppi;
- cout << *pi;
- cout << ppi;
- cout << *ppi;

Question 6

1 / 1 pts

Which of these is the preferred way to initialize a pointer so that it points to "nothing"?

- All are equally preferred.
- int *pi = nullptr;
- double *pd = 0;
- vector<int> *vp(NULL);
- Star *ps = NULL;

Question 7

1 / 1 pts

What is printed when you run this code?

```
int n{};
int *p = &n;
*p = 10;
n = 20;
cout << *p << endl;
```

- The address of n
- 10
- 20
- None of these
- 0

Question 8

1 / 1 pts

What is true about an uninitialized pointer?

- Dereferencing it will cause a program crash
- Dereferencing it is undefined behavior
- It is set to the `nullptr` value
- Dereferencing it is safe, but has no effect.
- None of these are true

Question 9

1 / 1 pts

Which expression obtains the value that ***p*** points to?

```
int x(100);
int *p = &x;
```

- ○ `&(*p)`
- ○ `p`
- ○ `&p`
- ◉ `*p`
- ○ `*(&p)`

Question 10

1 / 1 pts

Assume that ***ppi*** correctly points to ***pi***. Which line prints the address of ***ppi***?

```
int main()
{
    double pi = 3.14159;
    double *ppi;
    // code goes here
    // code goes here
}
```

- ○ `cout << *ppi;`
- ◉ `cout << &ppi;`
- ○ `cout << &pi;`
- ○ None of these
- ○ `cout << ppi;`

Question 11

1 / 1 pts

What is printed when you run this code?

```
int num = 0;
int *ptr = &num;
num = 5;
*ptr += 5;
cout << num << " " << *ptr << endl;
```

- ◉ `10 10`
- ○ `5 5`
- ○ `10 5`
- ○ Undefined; none of these
- ○ `5 10`

Question 12

1 / 1 pts

Assume that ***p1*** is a pointer to an integer and ***p2*** is a pointer to a second integer. Both integers appear inside a large contiguous sequence in memory, with ***p2*** storing a larger address. How many total integers are there in the slice between ***p1*** and ***p2***?

- ○ None of these
- ○ `p1 - p2;`
- ○ `p2 - p1 - 1;`
- ◉ `p2 - p1;`
- ○ `p1 - p2 + 1;`

Question 13

1 / 1 pts

Here is the pseudocode for the ***greenScreen()*** function from your homework. What single statement sets the red, green and blue components to **0**?

```
Let p point the beginning of the image
Set end to point just past the end
While p != end
    If *(p + 3) is 0 (transparent)
        Clear all of the fields
    Increment p by 4
```

- ○ `&(p + 1) = &(p + 2) = &(p + 3) = 0;`
- ○ `p = p + 1 = p + 2 = 0;`
- ◉ `*(p) = *(p + 1) = *(p + 2) = 0;`
- ○ None of these

- `*(p + 1) = *(p + 2) = *(p + 3) = 0;`

Question 14

1 / 1 pts

Here is a fragment of pseudocode for the ***negative()*** function from your homework. What statement represents the underlined portion of code?

```
Let p point to beginning of the image
Let end be pixel one past the end of the image
While p != end
    Invert the red component
    Move p to next component
```

- `*p++;`
- None of these
- `p++;`
- `*p = p + 1;`
- `&p++;`

Question 15

1 / 1 pts

Assume that p is a pointer to the first of 50 contiguous integers stored in memory. What is the address of the first integer appearing after this sequence of integers?

- `&p + 50;`
- None of these
- `p + sizeof(int) * 50;`
- `sizeof(p) + 50;`
- `p + 50;`