Review Test Submission: Quiz 1: Pointers

User	Phong Vo
Course	UML Computing II - Sec MW1 SU19 JMwaura
Test	Quiz 1: Pointers
Started	7/12/19 10:28 AM
Submitted	7/12/19 10:30 AM
Due Date	7/12/19 11:30 PM
Status	Completed
Attempt Score	17 out of 17 points
Time Elapsed	2 minutes out of 30 minutes
Results Displayed	All Answers, Submitted Answers, Correct Answers, Feedback, Incorrectly Answered Questions

Question 1

2 out of 2 points

State the equivalent pointer expression which refer the given array element a[i][j][k][l]?

Selected Answer: *(*(*(*(a+i)+j)+k)+l)

Correct Answer:

Evaluation Method	Correct Answer	Case Sensitivity
Contains	*(*(*(*(a+i)+j)+k)+l)	
Contains	*(*(*(*(a + i) + j) + k) +l)	
Contains	*(*(*(*(a + i) + j) + k) +l)	

Question 2

1 out of 1 points

What is the output of the following code ?

```
1.  #include <stdio.h>
2.      void pFunction(int*);
3.      int main()
4.      {
5.          int i = 10;
6.          pFunction((&i)++);
7.      }
8.      void pFunction(int *p)
9.      {
10.         printf("%d\n", *p);
11.     }
```

Selected Answers: c. Compile time error

Answers:
a. 10
b. Garbage value
c. Compile time error
d. Segmentation fault/code crash

Response Feedback: `void pFunction(int *p) does not return anything, so trying to increment (++) will not work.`

Question 3

1 out of 1 points

What is the output of the following C code?

```
1.  #include <stdio.h>
2.  void pFunction(int **p);
3.  int main()
4.  {
5.    int i = 97, *p = &i;
6.    pFunction(&p);
7.    printf("%d ", *p);
8.    return 0;
9.    }
10. void pFunction(int **p)
11. {
```

```
12.  int j = 2;
13.  *p = &j;
14.   printf("%d ", **p);
15. }
```

Selected Answer: a. 2 2

Answers: a. 2 2

b. 2 97

c. Undefined behaviour

d. Segmentation fault/code crash

Response Feedback: int **p passes the address of the pointer, so the value of *p gets changed by j

Question 4

1 out of 1 points

What is the output of the following C code?

```
1.    #include <stdio.h>
2.  void pFunction(int *const *p);
3.    int main()
4.     {
5.         int i = 11;
6.         int *p = &i;
7.         pFunction(&p);
8.         printf("%d ", *p);
9.     }
10.     void pFunction(int *const *p)
11.     {
12.         int j = 10;
13.         *p = &j;
14.         printf("%d ", **p);
15.     }
```

Selected Answer: a. Compile time error

Answers: a. Compile time error

b. 10 10

c. 10 11

d. Undefined behaviour

Response Feedback: const means that passed object cannot be modified

Question 5

1 out of 1 points

Combine the following two statements into one?

char *p;

p = (char*)malloc(100);

Selected Answer: char* p = (char*)malloc(100);

Correct Answer:

Evaluation Method	Correct Answer	Case Sensitivity
Contains	char* p = (char*)malloc(100);	
Contains	char* p = (char*) malloc(100);	
Contains	char * p = (char*) malloc(100);	
Contains	char * p = (char*)malloc(100);	
Contains	char *p = (char*)malloc(100);	
Contains	char *p=(char*)malloc(100);	
Exact Match	char* p = (char*)malloc(100);	
Exact Match	char* p= (char*)malloc(100);	

Question 6

1 out of 1 points

Use this question to answer the remainder of the questions.

Evaluate the following expressions assuming 32 bit integers and 32 bit pointers. Variables are declared as listed but after some unknown number of operations the current state of the memory is given by the supplied memory diagram.

```
struct my_vector
{
	int size;
	int capacity;
	int* data;
};
typedef struct my_vector My_vector;
My_vector v;
My_vector* p;
```

Variable Name / Address		**Memory Value**	
v	8000	3	v.size
	8004	4	v.capacity
	8008	9004	v.data contains 9004 as an address
p	8012	9028	
	8016	10000	
	8020	9020	
	…	…	
	9000	42	
	9004	63	v.data[0]
	9008	5	v.data[1]
	9012	100	v.data[2]
	9016	87	p->data[0]
	9020	14	p->data[1]
	9024	101	p->data[2]
	9028	2	p->size p->data[3]
	9032	3	p->capacity
	9036	9016	p->data

Questions:

1. v.data = 9004
2. (v.data[2]) << 2 = 100*4 = 400
3. &p = address of p = 8012
4. p->data[1] = 14
5. (*p).capacity = p->capacity = 3
6. p->data[3] % 5 = 2 % 5 = 2
7. (*p).size = 2

The shown data structure represents?

Selected Answer: vector

Correct Answer:

Evaluation Method	Correct Answer	Case Sensitivity
Exact Match	vector	
Exact Match	dynamic array	

Question 7

2 out of 2 points

```
v.data;
```

Selected Answer: 9004

Correct Answer:

Evaluation Method	Correct Answer	Case Sensitivity
Exact Match	9004	

Question 8

2 out of 2 points

```
(v.data[2]) << 2;
```

Selected Answer: 400

Correct Answer:

Evaluation Method	Correct Answer	Case Sensitivity
Exact Match	400	

Question 9

2 out of 2 points

&p;

Selected Answer: 8012

Correct Answer:

Evaluation Method	Correct Answer	Case Sensitivity
Exact Match	8012	

Question 10

2 out of 2 points

```
p->data[1];
```

Selected Answer: 14

Correct Answer:

Evaluation Method	Correct Answer	Case Sensitivity
Exact Match	14	

Question 11

2 out of 2 points

```
(*p).capacity;
```

Selected Answer: 3

Correct Answer:

Evaluation Method	Correct Answer	Case Sensitivity
Exact Match	3	

Friday, July 12, 2019 10:30:56 AM EDT

← OK