Talent Transformation (2019)

Started on	Tuesday, 14 August 2018, 10:44 AM
State	Finished
Completed on	Tuesday, 14 August 2018, 10:51 AM
Time taken	7 mins 16 secs
Grade	**5.00** out of 10.00 (**50**%)

Question 1

Correct

Mark 1.00 out of 1.00

⚑ Flag question

What will be the output of the program if the array begins 1200 in memory?

```c
#include<stdio.h>
int main()
{
int arr[]={2, 3, 4, 1, 6};
printf("%u, %u, %u\n", arr, &arr[0], &arr);
return 0;
}
```

Select one:
- ⦿ a. 1200, 1200, 1200 ✓
- ○ b. 1200, 1202, 1200
- ○ c. 1200, 1202, 1204
- ○ d. 1200, 1204, 1208

Explanation:

Step 1: int arr[]={2, 3, 4, 1, 6}; The variable arr is declared as an integer array and initialize

Step 2: printf("%u, %u, %u\n", arr, &arr[0], &arr); Here,

The base address of the array is 1200.

=> arr, &arr is pointing to the base address of the array arr.

=> &arr[0] is pointing to the address of the first element array arr. (ie. base address)

Hence the output of the program is 1200, 1200, 1200

The correct answer is: 1200, 1200, 1200

Question 2

Are the expressions arr and &arr same for an array of 10 integers?

Select one:

○ a. No

◉ b. Yes ✗

Explanation:

Both mean two different things. arr gives the address of the first int, whereas the&arr gives the address of array of ints.

The correct answer is: No

Which of the following statements are correct about 6 used in the program?
int num[6];
num[6]=21;

Select one:

○ a. In the first statement 6 specifies a particular element, whereas in the second statement it specifies a type.

◉ b. In the first statement 6 specifies a array size, whereas in the second statement it specifies a particular element of array. ✓

○ c. In the first statement 6 specifies a particular element, whereas in the second statement it specifies a array size.

○ d. In both the statement 6 specifies array size.

Explanation:

The statement 'B' is correct, because int num[6]; specifies the size of array andnum[6]=21; designates the particular element(7th element) of the array.

The correct answer is: In the first statement 6 specifies a array size, whereas in the second statement it specifies a particular element of array.

In C, if you pass an array as an argument to a function, what actually gets passed?

Select one:

○ a. First element of the array

○ b. Value of elements in array

○ c. Address of the last element of array

◉ d. Base address of the array ✓

Explanation:

The statement 'C' is correct. When we pass an array as a funtion argument, the base address of the array will be passe

The correct answer is: Base address of the array

What does the following declaration mean?
int (*ptr)[10];

Select one:

- ○ a. ptr is a pointer to an array of 10 integers
- ◉ b. ptr is array of pointers to 10 integers ✗
- ○ c. ptr is an pointer to array
- ○ d. ptr is an array of 10 integers

The correct answer is: ptr is a pointer to an array of 10 integers

---

Which of the following statements correct about k used in the below statement?
char ****k;

Select one:

- ○ a. k is a pointer to a char pointer
- ◉ b. k is a pointer to a pointer to a pointer to a pointer to a char ✓
- ○ c. k is a pointer to a pointer to a char
- ○ d. k is a pointer to a pointer to a pointer to a char

The correct answer is: k is a pointer to a pointer to a pointer to a pointer to a char

---

Will the following program give any warning on compilation in TurboC (under DOS)?
#include<stdio.h>
int main()
{
int *p1, i=25;
void *p2;
p1=&i;
p2=&i;
p1=p2;
p2=p1;

```
return 0;
}
```

Select one:

  ○ a. No

  ◉ b. Yes ✗

The correct answer is: No

Which of the statements is correct about the program?
```
#include<stdio.h>
int main()
{
int arr[3][3] = {1, 2, 3, 4};
printf("%d\n", *(*(*(arr))));
return 0;
}
```

Select one:

  ◉ a. Output: Garbage value ✗

  ○ b. Output: 3

  ○ c. Output: 1

  ○ d. Error: Invalid indirection

The correct answer is: Error: Invalid indirection

Point out the error in the program
```
#include<stdio.h>
int main()
{
int a[] = {10, 20, 30, 40, 50};
int j;
for(j=0; j<5; j++)
{
printf("%d\n", a);
a++;
}
return 0;
}
```

Select one:

  ○ a. Error: LValue required

b. Error: Rvalue required ✗

c. Error: Declaration syntax

d. Error: Expression syntax

The correct answer is: Error: LValue required

Question 10

Correct

Mark 1.00 out of 1.00

⚑ Flag question

Is there any difference between the following two statements?
char *p=0;
char *t=NULL;

Select one:

a. Yes

b. No ✓

Explanation:

NULL is #defined as 0 in the 'stdio.h' file. Thus, both p and t are NULL pointers.

The correct answer is: No

Finish review

QUIZ NAVIGATION

1 2 3 4 5 6 7 8 9 10

Show one page at a time

Finish review