Talent Transformation (2019)

Started on	Wednesday, 15 August 2018, 3:57 PM
State	Finished
Completed on	Wednesday, 15 August 2018, 4:09 PM
Time taken	11 mins 29 secs
Grade	**7.00** out of 10.00 (**70**%)

Question 1

Incorrect

Mark 0.00 out of 1.00

Flag question

Point out the error in the program?

```
#include<stdio.h>
int main()
{
struct a
{
float category:5;
char scheme:4;
};
printf("size=%d", sizeof(struct a));
return 0;
}
```

Select one:

- a. Error in this float category:5; statement
- b. None of above
- c. No error ✗
- d. Error: invalid structure member in printf

Explanation:

Bit field type must be signed int or unsigned int.
The char type: char scheme:4; is also a valid statement.

The correct answer is: Error in this float category:5; statement

Question 2

By default structure variable will be of auto storage class

Correct

Mark 1.00 out of 1.00

Flag question

Select one:

- a. Yes ✓
- b. No

The correct answer is: Yes

Question 3

Incorrect

Mark 0.00 out of 1.00

Flag question

Point out the error in the program?

```
#include<stdio.h>
int main()
{
struct emp
{
char name[25];
int age;
float bs;
};
struct emp e;
e.name = "Suresh";
e.age = 25;
printf("%s %d\n", e.name, e.age);
return 0;
}
```

Select one:

- a. No error, Output: Suresh 25 ✗
- b. Error: invalid constant expression
- c. Error: Rvalue required
- d. Error: Lvalue required/incompatible types in assignment

Explanation:

We cannot assign a string to a struct variable like e.name = "Suresh"; in
We have to use strcpy(char *dest, const char *source) function to assign a string.
Ex: strcpy(e.name, "Suresh");

The correct answer is: Error: Lvalue required/incompatible types in assignment

Question 4

Incorrect

Mark 0.00 out of 1.00

If a char is 1 byte wide, an integer is 2 bytes wide and a long integer is 4 bytes wide then will the following structure always occupy 7 bytes?

```
struct ex
{
char ch;
```

Flag question

```
int i;
long int a;
};
```

Select one:

- a. Yes ✗
- b. No

Explanation:

A compiler may leave holes in structures by padding the first char in the structure with another byte just to ensures that the integer that follows is stored at an location. Also, there might be 2extra bytes after the integer to ensure that the long integer is stored at an address, which is multiple of 4. Such alignment is done by machines to improve the efficiency of accessing values.

The correct answer is: No

Question 5

Correct

Mark 1.00 out of 1.00

Flag question

What will be the output of the program ?

```
#include<stdio.h>
int main()
{
enum status {pass, fail, absent};
enum status stud1, stud2, stud3;
stud1 = pass;
stud2 = absent;
stud3 = fail;
printf("%d %d %d\n", stud1, stud2, stud3);
return 0;
}
```

Select one:

- a. 0, 2, 1 ✓
- b. 1, 3, 2
- c. 1, 2, 3
- d. 0, 1, 2

The correct answer is: 0, 2, 1

Question 6

Correct

Which of the following statements are correct about the program below?

```
#include<stdio.h>
int main()
```

Mark 1.00 out of 1.00

Flag question

```
{
char str[20], *s;
printf("Enter a string\n");
scanf("%s", str);
s=str;
while(*s != '\0')
{
if(*s >= 97 && *s <= 122)
*s = *s-32;
s++;
}
printf("%s",str);
return 0;
}
```

Select one:

- a. The code converts upper case character to lower case
- b. The code converts a string in to an integer
- c. The code converts lower case character to upper case ✓
- d. Error in code

Explanation:

This program converts the given string to upper case string.
Output:
Enter a string: indiabix
INDIABIX

The correct answer is: The code converts lower case character to upper case

Question 7

Correct

Mark 1.00 out of 1.00

Flag question

What will be the output of the program ?

```
#include<stdio.h>
int main()
{
char str = "IndiaBIX";
printf("%s\n", str);
return 0;
}
```

Select one:

- a. IndiaBIX
- b. No output
- c. Error ✓
- d. Base address of str

Explanation:

The line char str = "IndiaBIX"; generates "Non portable pointer conversion" error.
To eliminate the error, we have to change the above line to
char *str = "IndiaBIX"; (or) char str[] = "IndiaBIX";
Then it prints "IndiaBIX".

The correct answer is: Error

Question 8

Correct

Mark 1.00 out of 1.00

Flag question

What will be the output of the program ?

```
#include<stdio.h>
#include<string.h>
int main()
{
static char s[] = "Hello!";
printf("%d\n", *(s+strlen(s)));
return 0;
}
```

Select one:

- a. Error
- b. 0 ✓
- c. 16
- d. 8

The correct answer is: 0

Question 9

Correct

Mark 1.00 out of 1.00

Flag question

What will be the output of the program ?

```
#include<stdio.h>
#include<string.h>
int main()
{
static char str1[] = "dills";
static char str2[20];
static char str3[] = "Daffo";
int i;
i = strcmp(strcat(str3, strcpy(str2, str1)), "Daffodills");
printf("%d\n", i);
return 0;
}
```

Select one:

- a. 4

b. 0 ✓

c. 1

d. 2

The correct answer is: 0

Question 10

Correct

Mark 1.00 out of 1.00

Flag question

If the size of pointer is 4 bytes then What will be the output of the program ?

```
#include<stdio.h>
int main()
{
char *str[] = {"Frogs", "Do", "Not", "Die", "They", "Croak!"};
printf("%d, %d", sizeof(str), strlen(str[0]));
return 0;
}
```

Select one:

a. 22, 4

b. 20, 2

c. 24, 5 ✓

d. 25, 5

Explanation:

Step 1: char *str[] = {"Frogs", "Do", "Not", "Die", "They", "Croak!"}; The variable str is declared as an pointer to the array of 6 strings.
Step 2: printf("%d, %d", sizeof(str), strlen(str[0]));
sizeof(str) denotes 6 * 4 bytes = 24 bytes. Hence it prints '24'
strlen(str[0])); becomes strlen(Frogs)). Hence it prints '5';
Hence the output of the program is 24, 5
Hint: If you run the above code in 16 bit platform (Turbo C under DOS) the output will be 12, 5. Because the pointer occupies only 2 bytes. If you run the above code in Linux (32 bit platform), the output will be 24, 5 (because the size of pointer is 4 bytes).

The correct answer is: 24, 5

Finish review

QUIZ NAVIGATION

1 2 3 4 5 6 7 8 9 10

Show one page at a time

Finish review

You are logged in as JEET SAHA 16900215019 (Log out)
ttc2019_2