Chapter 12 (ANSI C)

~~Roll~~ID 12102024 ;Reg: 10151

MULTIPLE CHOICE

12.1 (d) All of the above

12.2 (d) Both 1 and 2

12.3 (d) All of these

12.4 (a) * ptr

12.5 (a) A pointer variable stores NULL value until it is initialized.

REVIEW QUESTION

12.1 (a) True
(b) False
(c) False
(d) True
(e) True
(f) False
(g) True
(h) False
(i) True
(j) False

12.2 (a) address
(b) *
(c) *
(d) NULL
(e) 0
(f) passing by reference
(g) void type

12.3 A pointer is a variable that stores the memory address of an another variable.

To initialize we assign the memory address of an another variable.

```
int a=5, *b=&a;
```

12.4 (a) Here a is an integer variable and b is a pointer that points a.

(b) It is not correct. Because two declaration of p.

(c) Here s ~~ia~~ is a pointer of char data types.

(d) It is not correct. A parenthesis is missing. Correct form will be

```
a=((float *) &x);
```

It will cast the data of x into float type.

(e) It is a pointer of a function, where the return type is double.

12.5 (*m) [5] means m is a pointer of an array of 5 elements. Where *m[5] means array of pointers which can point 5 variables.

12.6 (a) (*p1)++ = 10

(b) --(*p2) = 9

(c) *p1 + (*p2)-- = 20

(d) ++(*p2) - *p1 = 1

12.7 For developing programs pointers are used in linked list, dynamic memory allocation and other algorithms.

12.8 Addition and increment or decrement operators are permitted for arithmetic operations on pointers.

12.9 100

12.10 (a) name + 10 will point the memory address after 10 element in the array.

(b) *(name+10) will return the value after 10 element in the array.

12.11 100

12.12 101 100

12.13 4

12.14 Both of those are equivalent. Because bot of them will except int ** m type variable, and can score double pointer or the address of single pointer.

12.15 No, they are not same. char s [5] will make an array of 5 characters while char *s will make an pointer which will point to a character's address.

12.16 (a) int (*P) (void);

DEBUGGING

12.1 (a) correct

(b) Incorrect. Here correct form is
p2 = &n ;

(c) Incorrect. Correct form is
*P1 = n ;

(d) Correct

(e) Correct

(f) Incorrect. Correct form is,

```
P1 = P2;
```

(g) Correct.

12.2 (a) Correct

(b) Incorrect. Here y is a pointer. So it will only accept address.

(c) Correct.

(d) Incorrret. Correct form will be,

```
int m;
int *x = &m;
```

12.3 Here, the address of a double pointer is assigned to a single pointer. This is why it won't work. Correct form will be,

```
int **P1, *P2;
P1 = &P2;
```

INTERVIEW QUESTIONS

12.1 Ans: For printing memory address %p is used.

12.2 Ans: A pointer to a pointer refers to a pointer which stores the memory address of an another pointer.

12.3 The output will be " 1 84 ". Here the first 1 is for * ptr which will mean the size of a pointer and the second value will depend on the architecture fact.

12.4 Null pointer is a pointer which will point to a null character.

12.5 Function pointer will point toward a function.

12.6 Void pointer is a pointer which can store the address of any data type.

12.7 Uninitilized pointer will point a random address where NULL has a constant 0.