Arrays and Pointers

CS201

Array

int a[4]

23		12		36		44	
a[0]		a[1]		a[2]		a[3]	
100	101	102	103	104	105	106	107
100		102		104		106	

‘a’ indicates address of a[0]

‘a’ = &a[0] (here a is 100 as per above table)

Therefore *a = a[0]

Contd…

- a+1 is address of a[1] (here a is 102)
- Therefore a[i] = *(a + i)
- Suppose int *p
- p = a
- p points to 100, p+3 points to 106

What's the difference between a and p?

- int a[4]
- a is a mnemonic
- p is a pointer variable
- a++
- a = a+1
- a = p

Cannot be done

Character Pointer

- char a[] = “CSEBTECH”(Note a is mnemonic)
- char *p = “CSEBTECH”

(Note p is not a mnemonic)

‘C’	‘S’	‘E’	‘B’	‘T’	‘E’	‘C’	‘H’
a[0]	a[1]	a[2]	a[3]	a[4]	a[5]	a[6]	a[7]

a[3]=M; (B is replaced by M)

p[3] = M; (B is replaced by M)

Example 1

1. What does the following fragment of C-Program print?

```
char c[] = “GATE2011”
char *p = c;
printf(“%s”, p+p[3]-p[1]);
```

A) GATE2011
B) E2011
C) 2011
D) 011

- Ans: Option “C”

Example2

2. printf(“%d”, printf(“ravi”));

- Answer: ravi4

Example 3

1. What does the following fragment of C-Program print?

```
char c[] = “GATE2018”
char *p = c;
printf(“%c%c”, *p, *(p+p[3]-p[1]));
```

A) G,2
B) G, k
C) GATE2018
D) None of these

- Option “A”

Example 4

- Which of the following c code snippet is not valid?

A) Char *p = “String1”; printf(“%c”,*++p);

B) Char q[]= “String1”; printf(“%c”,*++q);

C) Char *r= “String1”; printf(“%c”, r[1]);

D) None of the above

Array of Pointers

```
char *name[ ]= {“ramesh”, “raj”}

printf(“%s”, *(name + 1));
printf(“%s”, *(name) + 1);
printf(“%s”, (*(name+1) + 1));
printf(“%c”, name[1][1]);
```

One way of understanding

Solution:

name[0] contains address of ramesh.

name[1] contains address of raj.

name[1] = *(name+1);

printf(“%s”, *(name + 1)); Ans: raj

printf(“%s”, *(name) + 1); Ans: amesh

printf(“%s”, (*(name+1) + 1)); Ans: aj

printf(“%c”, name[1][1]); Ans: a

Another way of understanding

- char *name[ ]= {“ramesh”, “raj”}
- char name[2][8] = {“ramesh”, “raj”}

0	**r**	**a**	**m**	**e**	**s**	**h**	**\0**	
1	r	a	j	\0				
	0	1	2	3	4	5	6	7

Note: Multidimensional array and array of pointers are same.

Example2: Array of pointers

```
char *name[ ]= {“Madhu”, “Madhav”, “Madhubabu”}

   printf(“%s”, *(name + 1));
   printf(“%s”, *name+ 1);
   printf(“%s”, (*(name+2) + 7));
   printf(“%c”, name[2][7]);
```

Note: a[i][j] = *(*(a+i)+j)

char *name[ ]= {“Madhu”, “Madhav”, “Madhubabu”}

name

0	200
1	300
2	400

M	a	d	h	u	\0				
M	a	d	h	a	v	\0			
M	a	d	h	u	b	a	b	u	\0

char name[3][9]= {“Madhu”, “Madhav”, “Madhubabu”}

0	**M**	**a**	**d**	**h**	**u**	**\0**				
1	M	a	d	h	a	v	\0			
2	M	a	d	h	u	b	a	b	u	\0
	0	1	2	3	4	5	6	7	8	9