

GATE 2014 – C Programming

Question 10 – Pointers, scanf() and printf()

Question

Consider the following program in C language:

```
#include <stdio.h>

main()
{
    int i;
    int *pi = &i;

    scanf("%d", pi);
    printf("%d\n", i+5);
}
```

Which one of the following statements is TRUE?

- (A) Compilation fails.
- (B) Execution results in a run-time error.
- (C) On execution, the value printed is 5 more than the address of variable i.
- (D) On execution, the value printed is 5 more than the integer value entered.

Correct Answer: (D)

Solution – Step by Step

Step 1: Variable Declaration

The statement **int i;** creates an integer variable i.

int *pi = &i; declares pi as a pointer to an integer and stores the address of i in pi.

Variable	What it contains
i	No value assigned yet
pi	Address of i

Step 2: Understanding scanf()

```
scanf("%d", pi);
```

For **%d**, scanf() needs the address of an integer variable. Since **pi = &i**, pi contains the address of i. Therefore, this statement is equivalent to:

```
scanf("%d", &i);
```

Suppose the user enters **10**. Then the entered value is stored in i:

```
i = 10;
```

Important: The value 10 is stored in i, not in pi. pi continues to represent the address of i.

Step 3: Understanding printf()

```
printf("%d\n", i + 5);
```

If **i = 10**, then:

```
i + 5 = 10 + 5 = 15
```

Therefore, **15** is printed.

Checking the Options

(A) Compilation fails.

✗ FALSE

The program compiles successfully.

(B) Execution results in a run-time error.

✗ FALSE

pi points to the valid integer variable i, so scanf() can correctly store the input.

(C) The value printed is 5 more than the address of variable i.

✗ FALSE

The program prints $i + 5$, not $\&i + 5$.

(D) The value printed is 5 more than the integer value entered.

✓ TRUE

If the input is x, then $i = x$ and the output is $x + 5$.

🧠 Simple Way to Remember

```
int *pi = &i;
scanf("%d", pi);
```

is equivalent to

```
scanf("%d", &i);
```

Because **pi** already contains the address of **i**, scanf() stores the entered number in i. The program then prints **i + 5**.

Input = x → i = x → Output = x + 5

