|  |
| --- |
| **Question 17**  **WRONG** |

Predict the output?

|  |  |  |
| --- | --- | --- |
| int fun(char \*str1)  {    char \*str2 = str1;    while(\*++str1);    return (str1-str2);  }    int main()  {    char \*str = "GeeksQuiz";    printf("%d", fun(str));    return 0;  } | | |
| A | 10 |
|  | 9 |
| C | 8 |
|  | Random Number |

[**C Input and Output**](https://www.geeksforgeeks.org/c-language-2-gq/input-and-output-gq/)[**GATE CS Mock 2018**](https://www.geeksforgeeks.org/gate-cs-mock-2018/)  
[**Discuss it**](https://www.geeksforgeeks.org/gate-gate-cs-mock-2018-question-55/)

**Question 17 Explanation:**

The function fun() basically counts number of characters in input string. Inside fun(), pointer str2 is initialized as str1. The statement while(\*++str1); increments str1 till ‘\0’ is reached. str1 is incremented by 9. Finally the difference between str2 and str1 is returned which is 9. Option (B) is correct.

|  |
| --- |
| **Question 18**  **CORRECT** |

What is output of following C - Code?

|  |  |  |
| --- | --- | --- |
| #include <stdio.h>  #include <stdarg.h>  int fun(int n, ...)  {      int i, j = 1, val = 0;      va\_list p;      va\_start(p, n);      for (; j < n; ++j)      {          i = va\_arg(p, int);          val += i;      }      va\_end(p);      return val;  }  int main()  {      printf("%dn", fun(4, 1, 2, 3));      return 0;  } | | |
| A | 3 |
| B | 5 |
|  | 6 |
| D | 10 |

[**C Input and Output**](https://www.geeksforgeeks.org/c-language-2-gq/input-and-output-gq/)[**GATE CS Mock 2018 | Set 2**](https://www.geeksforgeeks.org/gate-cs-mock-2018-set-2/)  
[**Discuss it**](https://www.geeksforgeeks.org/gate-gate-cs-mock-2018-set-2-question-35/)

**Question 18 Explanation:**

The function receives variable number of arguments as there are three dots after first argument.   The firs argument is count of all arguments including first.  The function mainly returns sum of all remaining arguments. See <http://www.geeksforgeeks.org/how-to-count-variable-numbers-of-arguments-in-c> for details.