

## Quiz-3

### Question 1

 Time: 00:00:09

What will be the output of the pseudocode?

```
#include <stdio.h>

void main()
{
    int a = 100;

    printf("%0 %x", a);

}
```

☐ a

☐ %x

☐ 100

☐ %0

☒ %x

Ans:

Invalid format for the conversion of octal value but the printf function will print the %x as no parameter is passed to it

## Question 2

 Time: 00:00:03

Observe the code carefully and choose the correct output of this code.

```
#include <stdio.h>

int main()
{
    typedef int score;

    score result = 0.00;

    printf("%d", result);

    return 0;
}
```

☐ 0.0


☐ 0.00

☐ 0.000000

☐ 0 (Zero)

The type of score variable is defined above the declaration of the variable. Although the float value is assigned to the variable it will print the integer part of the variable

### Question 3

 Time: 00:00:03

Observe the code carefully and choose the correct output of this code.

```
#include <stdio.h>

int main(){

    float m = 0.0;

    long int n = 10;

    printf("%d", sizeof(m) == sizeof(m+n));

    return 0;
}
```

☐ 10

☐ 10.0

☐ 10+ 0.0

☐ 0 (Zero)

☒ 0 (Zero)

It will return 0 as the size of long int and float will not be equal

## Question 4

 Time: 00:00:03

What will be the output of following pseudo code?

```
#include <stdio.h>

int main()
{
    int any = ' ' * 10;
    printf("%d", any);
    return 0;
}
```

☐ 10

☐ 120

☐ 320

☐ 160

☒ 320

Ascii value of space (' ') is 32 and  $32 * 10$  is 320

## Question 5

 Time: 00:00:02

Observe the code carefully and choose the correct output of this code.

```
#include <stdio.h>

int main()
{
    int tell = 5.0, scl = 1*10;

    do {
        scl /= tell;
    } while(tell--);

    printf ("%d\n", scl);

    return 0;
}
```

☐ Floating-point exception

☐ 10


☐ 5.0

☐ 10.0

☒ Floating-point exception

This will get Floating point exception as the number is once get converted to float and it will catch the exception and stop the code

## Question 6

 Time: 00:00:03

What will be the output of the following c code?

```
#include <stdio.h>

int main()
{
    int b = 80;
    char a= (char)b;
    int i=sizeof(a);
    printf("%d",i);
}
```

☐ 1

☐ 80

☐ 2

☐ 1

☒ 1

The given code is good to run, the output of the following code is 1 as the character variable occupies the memory of 1 in the storage.

## Question 7

 Time: 00:00:03

what will be the output of following c code?

```
#include <stdio.h>

int main()
{
    printf("%f", main);
    return 0;
}
```

☐ 0 (Zero)

☐ 0.000

☐ 0.00000

☐ 0.000000

☒ 0.000000

It will print 0.000000 as main is given as argument in float format

## Question 8

 Time: 00:00:03

What will be the output of the following pseudo-code for input 7 ?

```
1. Read the value of N.  
2. Set m=1,T=0  
3. If m >N  
4.   Go to line No. 9  
5. Else  
6.   T= T+m  
7.   m=m+1  
8.   Go to line no.3  
9. Display the value of T  
10.Stop
```

☐ 28

☐ 32

☐ 56

☐ 76

☒ 28



```

#include<iostream>

using namespace std;

int main()
{
    int N=7;

    int m=1,T=0;

    while(1)
    {
        if(m >N)
            break;

        else
        {
            T= T+m;

            m=m+1;
        }
    }

    cout<<T;

    return 0;
}

```

The above code is c++ representation of the pseudocode where the while loop contains two condition and runs till the value of m is greater than N"

## Question 9

 Time: 00:00:03

what will be the output of the following algorithm for Num=10?

```
#include <stdio.h>

int main()
{
    float i;

    i = 1;

    printf("%d", i);

    return 0;
}
```

☐ Garbage value

☐ 1


☐ (0)Zero

☐ Error

☒ Garbage value

It will throw a garbage value because it is initialized as float and the print format is int.

## Question 10

 Time: 00:00:06

Consider the following pseudocode.

```
#include <stdio.h>

int main()
{
    int a,b;

    a=15;

    b=25;

    for(int i=0;i<5;i++)
    {
        if(i%2==0)

            ++a;

        else

            ++b;

    }

    printf("%d %d",a,b);

}
```

What is the value of b at the end of the pseudocode?

☐ 15 25

☐ 18 27

☐ 27 18

☐ 25 15

☒ 18 27

when the value of i will be 0,2 and 4, the value of a will be incremented by 1  
and when the value of i will be 1 and 3, the value of b will be incremented by 1  
hence the final value of a will be 18 and b will be 27



## Pseudocode quiz 3

### ① Output of pseudocode

```
#include <stdio.h>
void main() {
    int a = 100;
    printf("%100 %x", a);
}
```

Analys: ① format string "%100 %x" is invalid

② %x expect argu with hexadecimal

Ans: No correct options.

### ② Output of code

```
#include <stdio.h>
int main() {
    typedef int score;
    score result = 0.0;
    printf("%d", result);
    return 0;
}
```

Ans: result is of type int  
Assigning 0.00 to int truncates decimal part  
printf("%d", result) prints 0



### ③ Output of code

```
#include <stdio.h>
int main() {
    float n = 0.0;
    long int n = 10;
    printf("%010d", sizeof(n) == sizeof(++));
    return 0;
}
```

Ans:-  $(++)$  is invalid

- Size of (float) is 4 byte
- Size of long (int) is 8 byte
- Comparison  $4 == 8$  evaluates 0 (false)

### ④ Output of pseudocode

```
#include <stdio.h>
int main() {
    int any = ' ' * 10;
    printf("%010d", any);
    return 0;
}
```

Ans:- ASCII value of space (' ') is 32  
 $32 * 10 = 320$   
 printf("%010d", any) prints 320

### ⑤ Output of code



```
#include <stdio.h>
int main() {
    int tell = 5, scl = 10;
    do {
        scl /= tell;
    } while (tell--);
    printf("%d\n", scl);
    return 0;
}
```

### Analysis:-

- tell is initialized to 5
- scl is initialized to 10
- $scl / tell \rightarrow 10 / 5 = 2$
- tell-- decrements tell to 4
- Next iteration:  $2 / 4 = 0$  (integer division)
- 3, 2, 1 result in 0 by division
- tell=0, division by zero causes floating point exception

### (6) Output of code

```
#include <stdio.h>
int main() {
    int b = 80;
    char a = (char)b;
    int i = sizeof(a);
    printf("%d\n", i);
}
```

Ans: a is char variable assigned value 80  
 sizeof(a) returns sizeof char (byte)



printf("%d", i) return 1

### ⑦ Output of code (c)

```
#include <stdio.h>
int main() {
    printf("%d", main);
    return 0;
}
```

- Ans:
- main is function pointer
  - using %d to print function pointer is undefined behaviour
  - print 0.0000 garbage 9 > 7

### ⑧ Pseudocode:-

```
Read N
set m = 1, T = 0
If m > N, goto line 9
Else T = T + m, m = m + 1
Repeat until m > N
Display T
```

$m = 3 \rightarrow 3 > 7$   
 $2 \quad T = 1 + 2$   
 $4 > 7$   
 $2 \quad 1 + 2 =$   
 $2 + 1 = 3$   
 $1 + 2 = 3$   
 $3 \quad 2 + 3 = 5$   
 $3 \quad 3 + 5 = 8$   
 $4 \quad 4 + 8 = 12$   
 $5 \quad 5 + 12 = 17$   
 $6 \quad 6 + 17 = 23$   
 $7 \quad 7 + 23 = 30$   
 $8 \quad 8 + 30 = 38$   
 $9 \quad 9 + 38 = 47$   
 $10 \quad 10 + 47 = 57$   
 $11 \quad 11 + 57 = 68$   
 $12 \quad 12 + 68 = 80$   
 $13 \quad 13 + 80 = 93$   
 $14 \quad 14 + 93 = 107$   
 $15 \quad 15 + 107 = 122$   
 $16 \quad 16 + 122 = 138$   
 $17 \quad 17 + 138 = 155$   
 $18 \quad 18 + 155 = 173$   
 $19 \quad 19 + 173 = 192$   
 $20 \quad 20 + 192 = 212$   
 $21 \quad 21 + 212 = 233$   
 $22 \quad 22 + 233 = 255$   
 $23 \quad 23 + 255 = 278$   
 $24 \quad 24 + 278 = 302$   
 $25 \quad 25 + 302 = 327$   
 $26 \quad 26 + 327 = 353$   
 $27 \quad 27 + 353 = 380$   
 $28 \quad 28 + 380 = 408$   
 $29 \quad 29 + 408 = 437$   
 $30 \quad 30 + 437 = 467$   
 $31 \quad 31 + 467 = 498$   
 $32 \quad 32 + 498 = 530$   
 $33 \quad 33 + 530 = 563$   
 $34 \quad 34 + 563 = 597$   
 $35 \quad 35 + 597 = 632$   
 $36 \quad 36 + 632 = 668$   
 $37 \quad 37 + 668 = 705$   
 $38 \quad 38 + 705 = 743$   
 $39 \quad 39 + 743 = 782$   
 $40 \quad 40 + 782 = 822$   
 $41 \quad 41 + 822 = 863$   
 $42 \quad 42 + 863 = 905$   
 $43 \quad 43 + 905 = 948$   
 $44 \quad 44 + 948 = 992$   
 $45 \quad 45 + 992 = 1037$   
 $46 \quad 46 + 1037 = 1083$   
 $47 \quad 47 + 1083 = 1130$   
 $48 \quad 48 + 1130 = 1178$   
 $49 \quad 49 + 1178 = 1227$   
 $50 \quad 50 + 1227 = 1277$   
 $51 \quad 51 + 1277 = 1328$   
 $52 \quad 52 + 1328 = 1380$   
 $53 \quad 53 + 1380 = 1433$   
 $54 \quad 54 + 1433 = 1487$   
 $55 \quad 55 + 1487 = 1542$   
 $56 \quad 56 + 1542 = 1598$   
 $57 \quad 57 + 1598 = 1655$   
 $58 \quad 58 + 1655 = 1713$   
 $59 \quad 59 + 1713 = 1772$   
 $60 \quad 60 + 1772 = 1832$   
 $61 \quad 61 + 1832 = 1893$   
 $62 \quad 62 + 1893 = 1955$   
 $63 \quad 63 + 1955 = 2018$   
 $64 \quad 64 + 2018 = 2082$   
 $65 \quad 65 + 2082 = 2147$   
 $66 \quad 66 + 2147 = 2213$   
 $67 \quad 67 + 2213 = 2280$   
 $68 \quad 68 + 2280 = 2348$   
 $69 \quad 69 + 2348 = 2417$   
 $70 \quad 70 + 2417 = 2487$   
 $71 \quad 71 + 2487 = 2558$   
 $72 \quad 72 + 2558 = 2630$   
 $73 \quad 73 + 2630 = 2703$   
 $74 \quad 74 + 2703 = 2777$   
 $75 \quad 75 + 2777 = 2852$   
 $76 \quad 76 + 2852 = 2928$   
 $77 \quad 77 + 2928 = 3005$   
 $78 \quad 78 + 3005 = 3083$   
 $79 \quad 79 + 3083 = 3162$   
 $80 \quad 80 + 3162 = 3242$   
 $81 \quad 81 + 3242 = 3323$   
 $82 \quad 82 + 3323 = 3405$   
 $83 \quad 83 + 3405 = 3488$   
 $84 \quad 84 + 3488 = 3572$   
 $85 \quad 85 + 3572 = 3657$   
 $86 \quad 86 + 3657 = 3743$   
 $87 \quad 87 + 3743 = 3830$   
 $88 \quad 88 + 3830 = 3918$   
 $89 \quad 89 + 3918 = 4007$   
 $90 \quad 90 + 4007 = 4097$   
 $91 \quad 91 + 4097 = 4188$   
 $92 \quad 92 + 4188 = 4280$   
 $93 \quad 93 + 4280 = 4373$   
 $94 \quad 94 + 4373 = 4467$   
 $95 \quad 95 + 4467 = 4562$   
 $96 \quad 96 + 4562 = 4658$   
 $97 \quad 97 + 4658 = 4755$   
 $98 \quad 98 + 4755 = 4853$   
 $99 \quad 99 + 4853 = 4952$   
 $100 \quad 100 + 4952 = 5052$

Ans: Loop runs for 2-1 to 7 (Since 2)  
 in each iteration  $T + 7 \rightarrow T = 7 \times 7$   
 $= 49$

Out: 49

### ⑨ Algorithm (Pseudocode)



```
// include <Stdio.h>
int main() {
    float i = 1;
    printf("%d", i);
    return 0;
}
```

Ans - i is float initialized to 1.0  
 printf("%d", i) interpret float as int  
 leading to garbage value.

### ⑩ Value of b at End

Ans - 123 - -0 is always false, + + here  
 Cuts in every iteration

Loops runs 5 times:  $b = 25 + 5 = 30$

a remains 15

Correction:- Answer Suggest  $a = 18, b = 27$ , So, Out is 15 30 (bcq logic  
 can't support)