**Question 1**

**The output of the code below is ?**

#include <stdio.h>

int a;

void main()

{

if (a)

printf(“Hello”);

else

printf(“world”);

}

Hello

world

compile time error

none

**Answer :: (B)**  
  
**EXPLANATION :**  
  
It will print **"world"** as the value of a will be automatically initialized to zero(0) since it is a global variable so the**if condition will fail** and our program will print the else part,**i.e world.**

**Question 2**

**The output of the code below is**

#include <stdio.h>

void main()

{

int a = 5;

if (true);

printf(“hello”);

}

It will display hello

It will throw an error

No Output

Depends on Compiler

**Answer : (B)**  
  
**EXPLANATION :**  
  
The program will throw a syntax error as the if condition ends with an semi-colon which is a wrong syntax.

**Question 3**

**The output of the code below is**

#include <stdio.h>

void main()

{

int a = 0;

if (a == 0)

printf(“hi”);

else

printf(“how are u”);

printf(“hello”);

}

hi

how are you

hello

hihello

**Answer :: (D)**  
EXPLANATION :  
  
Since the value of a is 0 so the if condition will become true and print the "hi" after that at last "hello" will be printed.  
  
Hence the output : hihello

**Question 4**

**The following code ‘for(;;)’ represents an infinite loop. It can be terminated by.**

break

exit(0)

abort()

all of the mentioned

**Answer :: (A)**  
  
**EXPLANATION** **:**  
  
If we write the break statement inside the infinite loop it will terminate the infinite loop.

**Question 5**

**The correct syntax for running two variable for loop simultaneously is.**

a) for (i = 0; i < n; i++) for (j = 0; j < n; j += 5)

for (i = 0, j = 0;i < n, j < n; i++, j += 5)

for (i = 0; i < n;i++){}

for (j = 0; j < n;j += 5){}

**ANSWER :: (B)**  
  
**EXPLANATION :**  
  
The option (B) is the correct statement for writing two variable loop simultaneously.

**Question 6**

**Which for loop has range of similar indexes of ‘i’ used in**  
  
**for (i = 0;i < n; i++) ?**

for (i = n; i>0; i–)

for (i = n; i >= 0; i–)

for (i = n-1; i>0; i–)

for (i = n-1; i>-1; i–)

**Answer :(D)**  
  
**EXPLANATION :**

In the given for loop we are starting from 0 and going till n-1 so in this case the option d satisfies our range of similar indexes of "i" used in the loop.

**Question 7**

**The output of this C code is?**

#include <stdio.h>

void main()

{

int x = 0;

for (x < 3; x++)

{

printf(“Hello”);

}

}

Compile time error

Hello is printed thrice

Nothing

Varies

**Answer ::(A)**  
  
**EXPLANATION :**  
  
The given code snippet will throw an **compile time error** as the loop only contains two statement,  
  
Correct Syntax : for(;x<3;x++)  
  
This correct syntax will print "Hello" three times.

**Question 8**

**The output of this C code is?**

#include <stdio.h>

void main()

{

double x = 0;

for (x = 0.0; x < 3.0; x++)

printf(“Hello”);

}

Run time error

Hello is printed thrice

Hello is printed twice

Hello is printed infinitely

**Answer ::(B)**  
  
**EXPLANATION :**

**"Hello" will be printed thrice** as the value of x is declared double so it will  take the decimal values automatically(0.0) and run the loop three times to print "Hello" thrice.

**Question 9**

**The output of this C code is?**

#include <stdio.h>

int main()

{

do{

printf(“Inside while loop “);

}while (0);

printf(“Outside loop\n”);

}

Inside while loop

Inside while loop Outside loop

Outside loop

Infinite loop

**Answer ::(B)**  
  
**EXPLANATION :**

As we know that do-while is an exit controlled loop so the condition will be checked at the time of exit and "Inside while loop" will be printed and outside the loop "Outside loop" will be printed.

**Question 10**

**The output of this C code is?**

#include <stdio.h>

int main()

{

int i = 0;

do {

i++;

printf(“Inside while loop\n”);

} while (i < 3);

}

Inside while loop

Outside while loop

Depends on the compiler

Compile time error

Answer :(A)  
  
EXPLANATION :  
  
Inside while loop will be printed thrice as we know do-while is an exit controlled loop in which the condition is checked at the time of exit.