



Darshan Rahude <rahudedarshan@gmail.com>

Online Test - Biz4 Solutions, Pune |Campus Drive Organised by VibrantMinds Technologies,Pune.

1 message

Google Forms <forms-receipts-noreply@google.com>
To: rahudedarshan@gmail.com

Fri, May 28, 2021 at 2:35 PM

Google Forms

Thanks for filling out [Online Test - Biz4 Solutions, Pune |Campus Drive Organised by VibrantMinds Technologies,Pune.](#)

Here's what we got from you:

Online Test - Biz4 Solutions, Pune
|Campus Drive Organised by
VibrantMinds Technologies,Pune.

Email *

rahudedarshan@gmail.com

Registration

First Name *

Darshan

Last Name *

Rahude
.....

Mobile No. *

9028692061
.....

Alternate Mobile No.

9067183931
.....

Email *

rahudedarshan@gmail.com
.....

SSC Percentage *

84.40
.....

HSC Percentage (If not done then type NA) *

71.07
.....

Diploma Percentage(If not done then type NA) *

NA
.....

Diploma Specialization: *

NA
.....

Degree Aggregate Percentage: *

75.24

Degree Specialization: *

Computer Engineering

Graduation College Name: *

MES College Of Engineering Pune

Post-Graduation Aggregate Percentage: *

0

Post-Degree Specialization: *

NA

Post-Graduation College Name: *

NA

Online Test

What will be output if you will compile and execute the following c code? *

```
#include<stdio.h>

int main(){

    int i=10;

    static int x=i;

    if(x==i)

        printf("Equal");

    else if(x>i)

        printf("Greater than");

    else

        printf("Less than");

    return 0;

}
```

- ☒ Equal
- ☐ Greater than
- ☐ Less than
- ☐ Compiler error
- ☐ None of above

What is the output of this C code? *

```
#include <stdio.h>

void main()

{

    int b = 5 & 4 | 6;

    printf("%d", b);

}
```

- ☒ 6
- ☐ 4
- ☐ 1
- ☐ 0

What is the output of this C code? *

```
#include <stdio.h>

void main()
{
    char *a[10] = {"hi", "hello", "how"};
    int i = 0, j = 0;
    a[0] = "hey";
    for (i = 0; i < 10; i++)
        printf("%s\n", a[i]);
}
```

- ☐ hi hello how Segmentation fault
- ☐ hi hello how followed by 7 null values
- ☒ hey hello how Segmentation fault
- ☐ Depends on compiler

goto can be used to jump from main to within a function *

- ☐ true
- ☒ false
- ☐ Depends
- ☐ Varies

What is the output of this C code? *

```
#include <stdio.h>

int main()
{
    printf("%d ", 1);
    goto l1;
    printf("%d ", 2);
}

void foo()
{
    l1: printf("3 ", 3);
}
```

- ☐ 1 2 3
- ☐ 1 3
- ☐ 1 3 2
- ☒ Compile time error

Predict the output of following program. *

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
int a=10;
```

```
int b=2;
```

```
int c;
```

```
c=(a & b);
```

```
printf("c= %d",c);
```

```
return 0;
```

```
}
```

☐ c= 12

☐ c= 10

☒ c= 2

☐ c= 0

Consider the given statement: *

```
int x = 10 ^ 2
```

What will be the value of x?

☐ 5

☐ 6

☐ 7

☒ 8

What will be the output of following program ? *

```
#include <stdio.h>
```

```
int main()
```

```
{
```

```
    int x=2.3;
```

```
    const char c1=(float)x;
```

```
    const char c2=(int)x;
```

```
    printf("%d,%d\n",c1,c2);
```

```
    return 0;
```

```
}
```

- ☐ Error
- ☐ 2.3,2
- ☐ 2.300000,2
- ☒ 2,2

What will be the output of following program ? *

```
#include <stdio.h>
```

```
#define OFF 0
```

```
#if debug == OFF
```

```
    int a=11;
```

```
#endif
```

```
int main()
```

```
{
```

```
    int b=22;
```

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

```
    return 0;
```

```
}
```


☒ 11...22

☐ Error

☐ 11...11

☐ 22...22

*

```
#include <stdio.h>
char* strFun(void)
{
    char *str="IncludeHelp";
    return str;
}
int main()
{
    char *x;
    x=strFun();
    printf("str value = %s",x);
    return 0;
}
```

☒ str value = Garbage value

☐ str value = IncludeHelp

☐ Error

☐ No output

Which are the Access Modifiers in C++ class? *

☐ private

☐ public

☐ protected

☐ friend

☒ 1, 2 and 3

Which of the following statement is correct? *

- ☐ C++ enables to define functions that take constants as an argument.
- ☐ We cannot change the argument of the function that that are declared as constant.
- ☒ Both A and B.
- ☐ We cannot use the constant while defining the function.

Which of the following concepts means determining at runtime what method to invoke?

- ☐ Data hiding
- ☐ Dynamic Typing
- ☒ Dynamic binding
- ☐ Dynamic loading

Which of the following approach is adapted by C++? *

- ☐ Top-down
- ☒ Bottom-up
- ☐ Right-left
- ☐ Left-right

Which of the following statement is correct? *

- ☐ A reference is stored on heap.
- ☒ A reference is stored on stack.
- ☐ A reference is stored in a queue.
- ☐ A reference is stored in a binary tree.

*

void start()**{****A a = new A();****B b = new B();****a.s(b);****b = null; /* Line 5 */****a = null; /* Line 6 */****System.out.println("start completed"); /* Line 7 */****}****When is the B object, created in line 3, eligible for garbage collection?**

- ☒ after line 5
- ☐ after line 6
- ☐ after line 7
- ☐ There is no way to be absolutely certain.

Which cannot directly cause a thread to stop executing? *

- ☐ Calling the SetPriority() method on a Thread object.
- ☐ Calling the wait() method on an object.
- ☒ Calling notify() method on an object.
- ☐ Calling read() method on an InputStream object.

What is not an advantage of stored procedures? *

- ☐ Greater security
- ☐ SQL can be optimized

- ☐ Code sharing
- ☒ Increased network traffic

What SQL structure is used to limit column values of a table? *

- ☒ The LIMIT constraint
- ☐ The CHECK constraint
- ☐ The VALUE constraint
- ☐ None of the above is correct.

The HAVING clause does which of the following? *

- ☒ Acts like a WHERE clause but is used for groups rather than rows.
- ☐ Acts like a WHERE clause but is used for rows rather than columns.
- ☐ Acts like a WHERE clause but is used for columns rather than groups.
- ☐ Acts EXACTLY like a WHERE clause.

Which of the following is the correct order of evaluation for the below expression?
*

$$Z=X+Y*Z/4\%2-1$$

- ☐ / % + - =
- ☒ * / % + - =
- ☐ / * % + - =
- ☐ * % / - + =

What will be the output of the program? *

```
#include<stdio.h>

#define MAN(x, y) ((x)>(y)) ? (x):(y)

int main()
{
    int i=10, j=5, k=0;

    k = MAN(++i, j++);

    printf("%d, %d, %dn", i, j, k);

    return 0;
}
```

- ☐ 10, 5, 6
- ☒ 11, 5, 11
- ☐ 11, 5, Garbage
- ☐ 12, 6, 12

Which of the following statements about virtual base classes is correct? *

- ☐ It is used to provide multiple inheritance
- ☒ It is used to avoid multiple copies of base class in derived class.
- ☐ It is used to allow multiple copies if base class in derived class.
- ☐ It allows private members of the base class to be inherited in the derived class.

If the Copy constructor receives its arguments by value, The copy constructor would *

- ☐ Call one argument constructor of the class
- ☐ work without any problem

- ☒ Call itself recursively
- ☐ Call zero argument constructor

What happens when a class parameterized constructors and having no default constructor is used in a program and we create an object that needs a zero argument constructor. *

- ☐ Compile – Time error
- ☐ Preprocessing error.
- ☐ Option 3
- ☒ Runtime exception.

Which one is suitable syntax for function template?

- ☐ Template <class T> return_type Function_Name(parameters)
- ☐ Template <typename T> return_type Function_Name(parameters)
- ☒ both 1 and 2
- ☐ None of these

Which of the following operators have left to right associativity? *

- I. ()**
- II. []**
- III. ***
- IV. Identifier**

- ☐ I only
- ☐ II and III
- ☒ I and II

☐ I, II, and IV

What will be the output of the following program? *

```
#include<stdio.h>
void main()
{
    int i;
    struct a
    {
        int arr[5];
    };
    struct a p={1,2,3,4,5},q;
    q=p;
    for(i=4;i>=0;--i)
    {
        printf("%d ",q.arr[i]);
    }
}
```

- ☐ 5 4 3 2 1
- ☐ 4 3 2 1
- ☒ Compiler error
- ☐ 1 2 3 4 5

What will be the output of the following program? *

```
typedef enum
{
    x1,
    x2,
    x3,
}
X;
int main()
{
    X* xptr = calloc(1,
    sizeof(unsigned int));
    *xptr = x2;
    printf("%d", *xptr);
    *xptr = x3;
    printf("%d", *xptr);
    return 0;
}
```

- ☒ 12
- ☐ Compiler error: pointer to const cannot be allocated
- ☐ Compiler error: converting int to const int
- ☐ Runtime error

What will be the output of the following program? *

```
int main( )
{
    float i = 10.1;
    if(++i < 11.1)
        printf("Awesome");
    else
        printf("Aweful");
    return 0;
}
```

- ☐ Awesome
- ☒ Aweful
- ☐ Awesome Aweful
- ☐ Aweful Awesome

[Create your own Google Form](#)

[Report Abuse](#)