CPP QUIZ 02 (13-12-2023)
Malloc(), New, Reference

1. What will be the output of the following C++ code?

```
#include<iostream>
using namespace std;
int main( void )
{
        int num1 = 10;
        int num2 = 20;
        int &num3 = num1;
        num3 = num2;
        ++ num2;
        cout<<"Num1 : "<<num1<<endl;
        cout<<"Num2 : "<<num2<<endl;
        cout<<"Num3 : "<<num3<<endl;
        return 0;
}
```

Answers

1. Num1 : 10 Num2 : 21 Num3 : 10
2. Compile time Error
3. Num1 : 20 Num2 : 21 Num3 : 20
4. Num1 : 10 Num2 : 21 Num3 : 20

2. During dynamic memory allocation in C++, what happens if a new operator fails to allocate memory?

Answers

1. It returns False
2. It returns NULL
3. Throws bad_alloc exception
4. None of these

3. Which of the following is true about new when compared with malloc.

1) new is an operator, malloc is a function
2) new calls constructor, malloc doesn't
3) new returns appropriate pointer, malloc returns void * and pointer needs to typecast to appropriate type.

Answers

1. Both 1 and 3 are true
2. Both 2 and 3 are true
3. Both 1 and 2 are true
4. All 1, 2 and 3 are true

4. Which of the following statements are correct in context of below code in C++?

```
#include<iostream>
#include<stdlib.h>
using namespace std;
int main( void )
{
    int *p = malloc(10);
}
```

Answers

1. It will allocate memory for 10 integer variables in the heap section consecutively.
2. Compile time error
3. It will allocate memory for one integer variable in the heap section and memory will be initialized with 10 value.
4. Runtime error

5.
```
#include<iostream>
using namespace std;
#define a1 a
int main()
{
    int a1=1000;
    int &ref=a;
    cout<<"ref="<<ref<<endl;
    return 0;
}
```

Answers

1. ref=1000
2. compile time error
3. run time error
4. none of above

6. What will be the output of the following C++ code?

```
 #include<iostream>
 using namespace std;
 int main(void)
 {
        int #
        int a=5;
        &num=a;
        cout<<num;
        return 0;
 }
```

Answers

1. 5
2. Segmentation fault
3. Runtime error
4. Compile time error

```
7. #include<iostream>
using namespace std;
namespace NPoint
{
    class Point
    {
        int xPosition;
        int yPosition;
        public:
        Point()
        {
            xPosition=100;
            yPosition=200;
            cout << "parameterless constructor called";
        }
        Point(int value)
        {
            xPosition=value;
            yPosition=value;
            cout << "parameterized constructor called";
        }
        Point(const Point &other)
        {
            xPosition=other.xPosition;
            yPosition=other.yPosition;
            cout << "copy constructor called";
        }
    };
}
using namespace NPoint;
int main()
{
   Point t1;
   Point *t2=&t1;
   return 0;
}
which constructor called on following line?
Point *t2=&t1;
Answers
1. parameterless constructor called
2. parameterized constructor called
3. copy constructor called
4. none of above
```

8. Which of the following statements is/are incorrect about references in C++?
1.Once reference is initialized, we can change its referent later.
2.Reference is an alias or another name given to the existing memory location/object.

Answers

1. Both 1 and 2 are incorrect
2. Only 1 is incorrect
3. Only 2 is incorrect
4. None of the above

9. Which of the following statements are correct in context of below code?

```
#include<iostream>
using namespace std;
int main( void )
{
   int *ptr = new int(3);
   return 0;
}
```

Answers

1. we are allocating memory for a single variable but memory will be initialized with value 3.
2. we are allocating memory for array of three integer variable
3. we are allocating memory for array of three integer variable but memory will be initialized with 0 value
4. None of the above

10. if you allocate memory for an array using a new operator.
`int *ptr=new int[5];`
what syntax should you use to delete that allocated memory?

Answers

1. delete ptr;
2. free(ptr)
3. delete[] ptr;
4. both delete ptr and free(ptr)

---------- Compiled by Utkarsh Singh ----------