

Code Drills

Question 1

```
#include <iostream>
using namespace std;
```

```
int main()
{
    // declare variables
    int var1 = 3;
    int var2 = 24;
    int var3 = 17;

    // print address of var1
    cout << "Address of var1: " << &var1 << endl;

    // print address of var2
    cout << "Address of var2: " << &var2 << endl;

    // print address of var3
    cout << "Address of var3: " << &var3 << endl;
}
```

Question 2

```
int* pointVar, var;
var = 5;
```

```
// assign address of var to pointVar pointer
pointVar = &var;
```

Question 3

```
int* pointVar, var;
var = 5;
```

```
// assign address of var to pointVar
pointVar = &var;
```

```
// access value pointed by pointVar  
cout << *pointVar << endl; // Output: 5
```

Question 3

// C++ Program to display address of each element of an array

```
#include <iostream>  
using namespace std;
```

```
int main()
```

```
{  
    float arr[3];
```

```
    // declare pointer variable  
    float *ptr;
```

```
    cout << "Displaying address using arrays: " << endl;
```

```
    // use for loop to print addresses of all array elements  
    for (int i = 0; i < 3; ++i)
```

```
    {  
        cout << "&arr[" << i << "] = " << &arr[i] << endl;  
    }
```

```
    // ptr = &arr[0]  
    ptr = arr;
```

```
    cout<<"\nDisplaying address using pointers: "<< endl;
```

```
    // use for loop to print addresses of all array elements  
    // using pointer notation
```

```
    for (int i = 0; i < 3; ++i)  
    {  
        cout << "ptr + " << i << " = " << ptr + i << endl;  
    }
```

```
    return 0;
}
```

Question 4

// C++ Program to insert and display data entered by using pointer notation.

```
#include <iostream>
using namespace std;
```

```
int main() {
    float arr[5];
```

```
    // Insert data using pointer notation
    cout << "Enter 5 numbers: ";
    for (int i = 0; i < 5; ++i) {
```

```
        // store input number in arr[i]
        cin >> *(arr + i) ;
```

```
    }
```

```
    // Display data using pointer notation
    cout << "Displaying data: " << endl;
    for (int i = 0; i < 5; ++i) {
```

```
        // display value of arr[i]
        cout << *(arr + i) << endl ;
```

```
    }
```

```
    return 0;
}
```

Question 5

// function that takes value as parameter

```
void func1(int numVal) {
```

```

    // code
}

// function that takes reference as parameter
// notice the & before the parameter
void func2(int &numRef) {
    // code
}

int main() {
    int num = 5;

    // pass by value
    func1(num);

    // pass by reference
    func2(num);

    return 0;
}

```

Question 6

```

#include <iostream>
using namespace std;

// function prototype with pointer as parameters
void swap(int*, int*);

int main()
{

    // initialize variables
    int a = 1, b = 2;

    cout << "Before swapping" << endl;
    cout << "a = " << a << endl;
    cout << "b = " << b << endl;
}

```

```

// call function by passing variable addresses
swap(&a, &b);

cout << "\nAfter swapping" << endl;
cout << "a = " << a << endl;
cout << "b = " << b << endl;
return 0;
}

// function definition to swap numbers
void swap(int* n1, int* n2) {
    int temp;
    temp = *n1;
    *n1 = *n2;
    *n2 = temp;
}

```

Question 7

```

#include <iostream>
using namespace std;

int main() {

    // declare an int pointer
    int* pointInt;

    // declare a float pointer
    float* pointFloat;

    // dynamically allocate memory
    pointInt = new int;
    pointFloat = new float;

    // assigning value to the memory
    *pointInt = 45;
    *pointFloat = 45.45f;

    cout << *pointInt << endl;
}

```

```

cout << *pointFloat << endl;

// deallocate the memory
delete pointInt;
delete pointFloat;

return 0;
}

```

Question 8
// C++ Program to store GPA of n number of students and display it
// where n is the number of students entered by the user

```

#include <iostream>
using namespace std;

int main() {

    int num;
    cout << "Enter total number of students: ";
    cin >> num;
    float* ptr;

    // memory allocation of num number of floats
    ptr = new float[num];

    cout << "Enter GPA of students." << endl;
    for (int i = 0; i < num; ++i) {
        cout << "Student" << i + 1 << ": ";
        cin >> *(ptr + i);
    }

    cout << "\nDisplaying GPA of students." << endl;
    for (int i = 0; i < num; ++i) {
        cout << "Student" << i + 1 << ": " << *(ptr + i) << endl;
    }
}

```

```

// ptr memory is released
delete[] ptr;

return 0;
}
Question 9
#include <iostream>
using namespace std;

class Student {
private:
    int age;

public:

    // constructor initializes age to 12
    Student() : age(12) {}

    void getAge() {
        cout << "Age = " << age << endl;
    }
};

int main() {

    // dynamically declare Student object
    Student* ptr = new Student();

    // call getAge() function
    ptr->getAge();

    // ptr memory is released
    delete ptr;

    return 0;
}

```