

PS4

- Don't wait till the last moment.
-

Q1

Write the definition for a class named `Vector2D` that stores information about a two-dimensional vector. The class should have methods to get and set the `x` component and the `y` component, where `x` and `y` are integers.

Next, overload the `*` operator so that it returns the dot product of two vectors. The dot product of two-dimensional vectors `A` and `B` is equal to $(A_x \times B_x) + (A_y \times B_y)$.

Use the following main function (Change test data. Create objects with different data):

```
// =====  
//      main function  
// =====  
int main()  
{  
    // Some test vectors  
    Vector2D v1(10,0), v2(0,10), v3(10,10), v4(5,4);  
  
    cout << "(" << v1.GetX() << "," << v1.GetY() << ") * (" << v2.GetX()  
        << "," << v2.GetY() << ") = " << v1*v2 << endl;  
    cout << "(" << v2.GetX() << "," << v2.GetY() << ") * (" << v3.GetX()  
        << "," << v3.GetY() << ") = " << v2*v3 << endl;  
    cout << "(" << v3.GetX() << "," << v3.GetY() << ") * (" << v4.GetX()  
        << "," << v4.GetY() << ") = " << v3*v4 << endl;  
    return 0;  
}
```

Q2

Define a class named `MyInteger` that stores an integer and has methods to get and set the integer's value. Then, overload the `[]` operator so that the index returns the digit in position `i`, where `i=0` is the least significant digit. If no such digit exists then `-1` should be returned.

For example, if `x` is of type `MyInteger` and it is set to `418`, then `x[0]` should return `8`, `x[1]` should return `1`, `x[2]` should return `4`, and `x[3]` should return `-1`.

Hint: Use `/` and `%` to extract a single digit from an integer. You might want to use the `pow` function to compute 10^i . The function is defined in [Appendix 4](#) and requires the `cmath` library.

Use the following main function (Change the object data. Use different integers):

```
// =====  
//      main function  
// =====  
int main()  
{  
    // Some test numbers  
    MyInteger num(418);  
  
    cout << num[0] << " " << num[1] << " " << num[2] << endl;  
    cout << num[3] << endl;  
    cout << num[-1] << endl;  
  
    return 0;  
}
```

Turn In

- Make and submit a zip file(<your_full_name>_PS4.zip) which includes the following:
 - Source code of Q1: q1.cpp
 - Source code of Q2: q2.cpp
 - Run Q1 and Q2 and attach screenshots(in jpg format, not exceeding 300kb each) which show that your programs are running.
 - At least 1 screenshot for each question.