OOPS LAB

PROGRAMMING ASSIGNMENT № 6

POLYMORPHISM: TYPE CONVERSION

CSE Department, JMI 26/09/2023

Read carefully before you begin:

- Total Marks: 30. Each question carries 10 marks.
- You have 2 hours to complete the assignment. Failure to have your program evaluated before you leave the lab will cause forfeiture of the grade for that lab.
- In order to receive full marks, you must demo the full working code and show the output and given an explanation of your approach where applicable.
- Please **save your code** throughout the semester in a place where you do not lose it. You will be required to submit it at the end.
- Use proper filenaming conventions and commenting. Code that is hard to read or understand will incur a penalty.
- Collaboration must kept to general discussions only. Please do NOT share code or directly share answers with each other. Plagiarism is unacceptable.
- Note: Your Point class must have all of the previous functionality from earlier lab assignments. If it does not, please spend time after today's lab to catch up.

Problem 1 : Basic Types and User-defined Types (10 marks)

1. Implement a parameterized constructor for your **Point** class that takes the starting address of an array as a pointer to a **double** and uses it to make a Point object.

   Eg: double arr[2] = {1,1}; Point p1(arr);

2. Now implement a conversion operator inside the **Point** class that returns a pointer to **double** that points to an array of size 2.

3. Let's say I have the following code, will it use the type conversion operator or the overloaded subscript operator?

Listing 1: Sample C++ code – streamt output in a 2D Point C++. (Source: Saif Ali)

```
#include <iostream>

```

```cpp
int main()
{
    double arr = {1,2};
    Point p1(arr);
    cout << p1[1] << "," << p1[2];

    double * parr = p1;
    cout << parr[1] << "," << parr[2];
}
```

Problem 2: Between User-defined Types (10 marks)

1. Define and implement a class **Vector** and provide all constructors and a default destructor.
2. Provide a conversion operators from **Point** to **Vector** and the other way round.
3. What is the output of the code? Explain why you get the output that you do.

   ```
   Vector v1;
   Point p1(1,1);
   v1 = p1;
   cout << v1 ;
   ```

Problem 3: Image loading and sampling. (10 marks)

1. Include the file **stb_image.h**.
2. Define and implement a class called **Image** that has one member as follows:

   std::vector<unsigned char> _data;

   This is a vector of unsigned chars that holds the image data.
3. Implement a parameterized constructor inside the **Image** class that takes a path (string) to an image file and loads it. The code for loading an image is given below.
4. Implement a member function called **sample** inside the class that takes a **Point** as an argument and returns the pixel value at the location of (_x,_y). Recall that these are double type.

Listing 2: Sample C++ code – loading an image. (Source: Muneeb Ahmad)

```
//passing 1 as required components since we only need one channel
unsigned char* data = stbi_load(path, &width, &height, &numComponents, 1);

if(data == NULL)
{
    //todo error
}

_data = std::vector<unsigned char>(data, data + width* height);

stbi_image_free(data);
```