LASALLE COLLEGE VANCOUVER

Function Pointers, Functors, lambdas, std::function

VGP 131 - Object Oriented Programming in C++ II

Instructor: Ivaldo Tributino

April 28, 2022

ASSIGNMENT INSTRUCTION

- The assignment must be submitted by May 01 30, 2022.
- Each problem presents its own score, the sum of all scores is 100.
- The Polygon class can be found in the appendix.

Student's Number:
Student's Name:

(10 POINTS) PROBLEM 1

What value must **a** take for the output to be 70?

```
int x = 3;

auto f1 = [&](int y){return x+=y;};

auto f2 = f1;

cout << f2(f1(a)) << endl;
```

(10 Points) Problem 2

Explain why the code does not compile and what must be added to make it work.

```
int main ()
{
    int a_1 = 0;

    auto test = [=](){
        return ++a_1;
    };

    int res = test();

    std::cout << res << std::endl;

    return 0;
}
```

(15 Points) Problem 3

Write a version of a basic calculator using function pointers.

(15 Points) Problem 4

Complete the code below to get the output shown in the next gray box. Take the time to explore std::pair, std::make_pair, and std::printf if you are not familiar with them.

```
using StrFuncContainer = std::vector<std::pair<string,std::function<double(
    float)>>>;

StrFuncContainer Polygons;

for(int i=3; i<9; i++){
  Polygon poly(i);
  Polygons.emplace_back(std::make_pair("write_your_code_here"));
};

for(auto const &pair : Polygons){
  cout << "----␣" << pair.first << "␣----" << ’\n’;
  for(int i =1; i<6; i++){
    std::printf("write_your_code_here");
  }
  cout << ’\n’;
}
```

Output:

```
---- triangle ----
Area(1) = 0.43 | Area(2) = 1.73 | Area(3) = 3.90 | Area(4) = 6.93 | Area(5)
     = 10.83 |
---- square ----
Area(1) = 1.00 | Area(2) = 4.00 | Area(3) = 9.00 | Area(4) = 16.00 | Area
    (5) = 25.00 |
---- pentagon ----
Area(1) = 1.72 | Area(2) = 6.88 | Area(3) = 15.48 | Area(4) = 27.53 | Area
    (5) = 43.01 |
---- hexagon ----
Area(1) = 2.60 | Area(2) = 10.39 | Area(3) = 23.38 | Area(4) = 41.57 | Area
    (5) = 64.95 |
---- heptagon ----
Area(1) = 3.63 | Area(2) = 14.54 | Area(3) = 32.71 | Area(4) = 58.14 | Area
    (5) = 90.85 |
---- octagon ----
Area(1) = 4.83 | Area(2) = 19.31 | Area(3) = 43.46 | Area(4) = 77.25 | Area
    (5) = 120.71 |
```

(15 Points) Problem 5

Make some changes to the previous code as shown below and comment if there was an improvement or a worsening in performance.

```
using StrFuncContainer = std::vector<std::pair<std::function<string()>,std
    ::function<double(float)>>>;

StrFuncContainer Polygons;
Polygon* polyptr = new Polygon(3);

Polygons.emplace_back(std::make_pair("write_your_code_here"));

for(unsigned int i=3; i<9;i++){
  polyptr->setNumberSides(i);
  cout << "----␣" << "write_your_code_here"<< "␣----" << ’\n’;
    for(int i =1; i<6; i++){
    std::printf("write_your_code_here");
  }
  cout << ’\n’;
}

delete polyptr;
```

(15 Points) Problem 6

Create a sort function to sort a vector<Polygon> v. Enable this function to receive the following functions as one of its parameters.(Hint: Overload the comparison operators)

```
bool ascending(Polygon x, Polygon y)
{
    return x > y;
}

bool descending(Polygon x, Polygon y)
{
    return x < y;
}

//Ex. ascending(triangle, square) is false.
```

(20 Points) Problem 7

Prove through an example that functors and virtual functions are closely related.