BYU
IDAHO

Intro to Exceptions

CS 165 – Object Oriented Software Development

Macbeth – Lesson 3.1

Agenda

- Opening Prayer
- Scripture
- Q&A
  - Review Project
- Nested Structs
- Dynamically Allocated Structs
- Looking Forward

BYU IDAHO

Scripture

Moroni 7:47-48

But charity is the pure love of Christ, and it endureth forever; and whoso is found possessed of it at the last day, it shall be well with him.

Wherefore … pray unto the Father with all the energy of heart, that ye may be filled with this love, which he hath bestowed upon all who are true followers of his Son, Jesus Christ; that ye may become the sons [and daughters] of God; that when he shall appear we shall be like him, for we shall see him as he is.

Updated Grading for Checkpoints

- Checkpoints are an auto-graded pass/fail homework.
- You will receive 4 points (100%) if you turn in the checkpoint (with testbed passing all tests) before the weekly assignment/project is due on Monday. Previous grades have already been updated.
- The reason for the change is to encourage you to use checkpoints to ponder the new material each week and successfully implement the weekly assignment/project.
- Please do not turn in assignments/projects late. You need to keep moving and start on the next topic.

Review Project

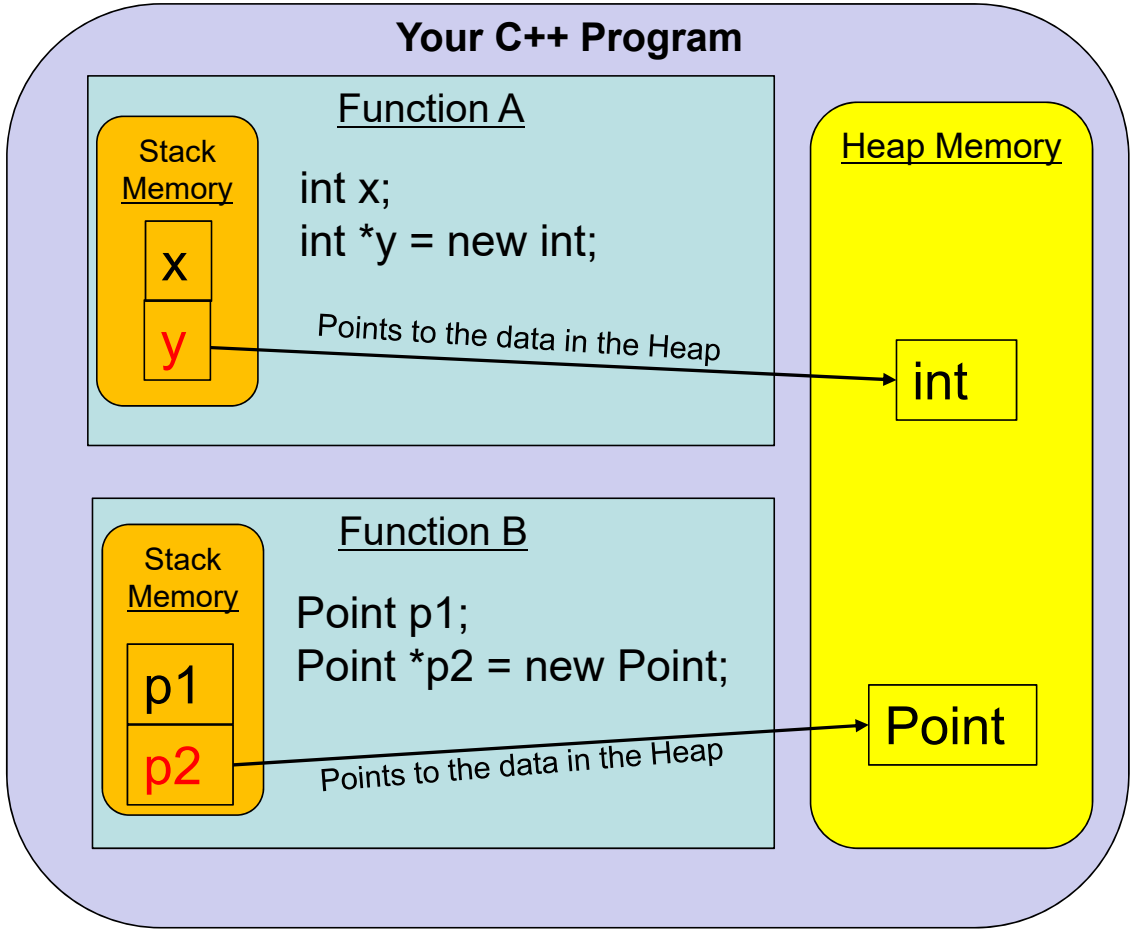
Points and Lines

```
struct Point
{
   double x;
   double y;
};


struct Line
{
   Point p1;
   Point p2;
   int color;
};
```

Stack and Heap

- Every function has stack memory which stores all your variables.
- If you use the “new” keyword, then the variable is stored in the heap.
- The function stack memory needs to maintain a pointer to the variable in the heap (a.k.a don’t lose it!)



Allocating Structures

	Statically Allocated Structure	Dynamically Allocated Structure
Where is it stored?	Stack Memory in the Function	Heap Memory in the Program
When is it stored?	At Compile Time	At Run Time

Example
 	
`void disp(Point *p)`
`{`
`  if (p == NULL)`
`  {`
`  return;`
`  }`
`  cout << p->x`
`  << " "`
`  << p->y;`
`}`
	`Point p1;`
	
	`p1.x = 2;`
	`p1.y = 3;`
	
	`disp(&p1);`
		`Point *p2 = new Point;`
		
		`p2->x = 2;`
		`p2->y = 3;`
		
		`disp(p2);`
		
		`delete p2;`

Error Handling

- How do I communicate to the calling function that something wrong has happened?

BYU IDAHO

Exceptions

```
void safeDivide(const double num, const double denom, double &result)
{
   if (denom == 0)
   {
      throw string("Can not divide by 0.");
   }
   result = num / denom;
}
```

BYU IDAHO

Exceptions

```
int main()
{
   double result = 0.0, numerator = 0.0, denominator = 0.0;

   cout << "Enter numerator and denominator : ";
   cin >> numerator >> denominator;
   try
   {
      safeDivide(numerator, denominator, result);
      cout << "Result = " << result << endl;
   }
   catch (const string err)
   {
      cout << "ERROR! " << err << endl;
   }
   return 0;
}
```

BYU IDAHO

Looking Forward

- Tuesday
  - Checkpoint 3A
- Wednesday
  - Checkpoint 3B
- Monday
  - Assignment 3 – Digital Forensics with Corrupt Files

BYU IDAHO