**ใบงานที่ 6**

**Static Library**

# **วัตถุประสงค์**

1. สร้าง Static Library
2. ใช้งาน Github
3. ใช้เครื่องมือต่างๆ ในการสร้าง Application/Library

# **เนื้อหาเบื้องต้น**

https://github.com/Desktop-Programming-Lab-2559/LAB-02/blob/master/LabIntro.md

# **ลำดับการทดลอง**

1. New Project (Name:MathFuncs | Type: Static Library)
2. New File (Name:MathFuncsLib.h)

|  |
| --- |
| // MathFuncsLib.h  namespace MathFuncs  {  class MyMathFuncs  {  public:  // Returns a + b  static double Add(double a, double b);  // Returns a - b  static double Subtract(double a, double b);  // Returns a \* b  static double Multiply(double a, double b);  // Returns a / b  static double Divide(double a, double b);  };  } |

1. New File (Name:MathFuncsLib.cpp)

|  |
| --- |
| // MathFuncsLib.cpp  // compile with: cl /c /EHsc MathFuncsLib.cpp  // post-build command: lib MathFuncsLib.obj  #include "MathFuncsLib.h"  #include <stdexcept>  using namespace std;  namespace MathFuncs  {  double MyMathFuncs::Add(double a, double b)  {  return a + b;  }  double MyMathFuncs::Subtract(double a, double b)  {  return a - b;  }  double MyMathFuncs::Multiply(double a, double b)  {  return a \* b;  }  double MyMathFuncs::Divide(double a, double b)  {  return a / b;  }  } |

1. Make Project
2. New Project(Name:MyExecRefsLib | Type:Console Application | Application Type: An Empty Project )
3. New File (Name:MyExecRefsLib.cpp)

|  |
| --- |
| // MyExecRefsLib.cpp  // compile with: cl /EHsc MyExecRefsLib.cpp /link MathFuncsLib.lib  #include <iostream>  #include "MathFuncsLib.h"  using namespace std;  int main()  {  double a = 7.4;  int b = 99;  cout << "a + b = " <<  MathFuncs::MyMathFuncs::Add(a, b) << endl;  cout << "a - b = " <<  MathFuncs::MyMathFuncs::Subtract(a, b) << endl;  cout << "a \* b = " <<  MathFuncs::MyMathFuncs::Multiply(a, b) << endl;  cout << "a / b = " <<  MathFuncs::MyMathFuncs::Divide(a, b) << endl;  return 0;  } |

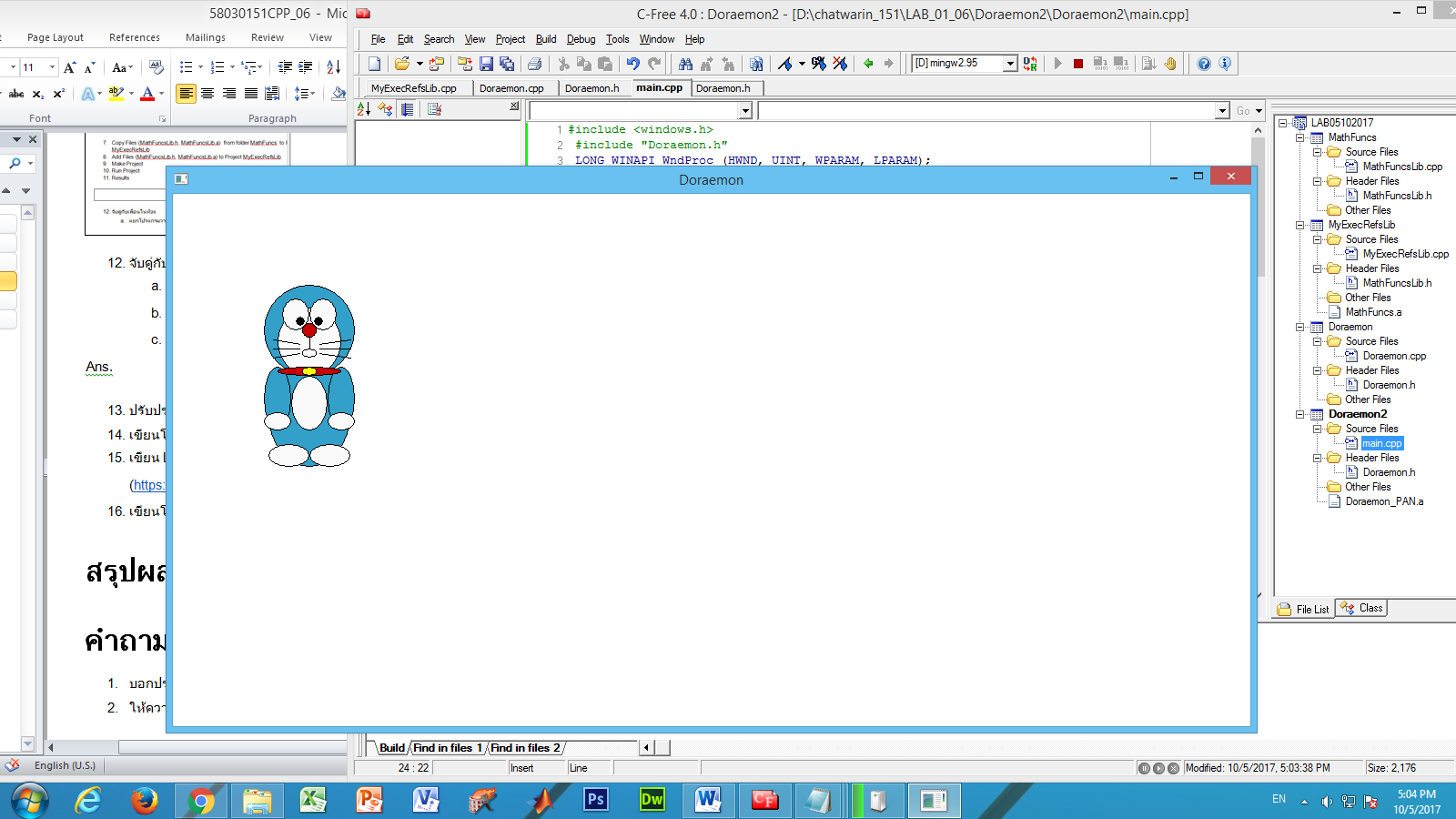
1. Copy Files (MathFuncsLib.h, MathFuncsLib.a) from folder MathFuncs to Folder MyExecRefsLib
2. Add Files (MathFuncsLib.h, MathFuncsLib.a) to Project MyExecRefsLib
3. Make Project
4. Run Project
5. Results

|  |
| --- |
| Ans. เป็นการสร้างไฟล์เก็บชุดคำสั่งใน static library เป็นการสร้างฟังก์ชั่นการคำนวณคณิตศาสตร์ไว้ให้เรียกใช้อีกที เป็นไฟล์ .h สร้าง namespace และ class ที่จะใช้ และ .cpp สร้างฟังก์ชั่นให้มีการคำนวณ ในการเขียนโค้ดในโปรเจค console application จะเห็นว่าโค้ดในโปรเจค console มีแค่กำหนดค่าตัวแปรแล้วเรียกใช้ function ในชุดคำสั่งที่สร้างไว้ |

1. จับคู่กับเพื่อนในห้อง
   1. แยกโปรแกรมวาด Doraemon ออกมาเป็น Library (Doraemon.a)
   2. เขียนคู่มือการใช้งาน Library

* ประกาศชื่อ library ที่ด้านบน ชื่อ Doraemon.h
* ประกาศตัวแปรใน library ซึ่งก็คือ HWND ก่อนนำไปใช้ ต้องกำหนด HWND = hwnd ในโปรแกรมให้ตรงกันทุกตัวก่อนจึงจะนำ library ไปใช้ได้
* ก่อนจะใช้ในส่วนของ case WM\_PAINT ต้องใส่ค่า namespace และ class ก่อน โดยใส่ค่าตามนี้
* namespace = Doraemon
* class = MyDoraemon
  1. เขียนโปรแกรมวาด Doraemon โดยใช้ Library

Ans. เป็นการเรียกใช้ library ของเพื่อน โดยใช้ไฟล์นามสกุล .a และ .h ของเพื่อนมา add ใส่ในโปรแกรมของเรา โดยต้องเช็คการใช้ชื่อให้ตรงกันด้วย



1. ปรับปรุงโปรแกรมที่ใช้ทดสอบ โดยแยกออกมาเป็น Library
2. เขียนโปรแกรมทดสอบ
3. เขียน Library การคำนวนเกี่ยวกับวงจรไฟฟ้า (ไม่ซ้ำกัน) ตั้งชื่อตามรหัสนักศึกษา Add ลง Github (<https://github.com/kjpiya/Electrical.git>) พร้อมใส่คำอธิบายลงใน README.MD
4. เขียนโปรแกรมคำนวนวงจรไฟฟ้า โดยมีฟังก์ชันอย่างน้อย 10 ฟังก์ชัน (ดึง Library ของเพื่อนมาใช้

# **สรุปผลการทดลอง**

Ans. จากการทดลองจะพบว่าเป็นการสร้าง library เพื่อสามารถนำไปใช้ได้ในงานที่ต้องทำกับคนหลายๆคนหรืองานที่มีการคำนวณหรือมีความซับซ้อนของ code เป็นการรวมชุดคำสั่งไว้ด้วยกัน ในการสร้าง library ต้องเริ่มจากการกำหนด namespace กับ class ให้ได้ว่ามีอะไรบ้าง ใน project type static library จากนั้นสร้าง project ใหม่ type conceal window หรือ window application เป็นการสร้างโปรเจคเพื่อเรียกใช้ library ที่เราทำไว้ได้ การสร้าง library ทำให้การพิมพ์โค้ดคำสั่งที่ซับซ้อนหรือต้องเรียกใช้ชุดคำสั่งนั้นหลายๆครั้งสะดวกขึ้น แต่ข้อควรระวังคือการตั้งชื่อเรียกใช้อาจจะตั้งไม่ตรงกันหรือการกำหนดชนิดของข้อมูลผลลัพธ์ไม่ถูกต้อง

# **คำถามท้ายการทดลอง**

1. บอกประโยชน์ของการใช้ Library

Ans. - หลังจากการ Compile แล้ว สามารถให้คนอื่นเอาไปใช้ต่อ

- จัดเก็บชุดคำสั่งไว้ที่เดียว สามารถเรียกใช้งานสะดวกขึ้น

1. ให้ความหมาย Dynamic-link library

Ans. Dynamic=link library หรือ DLL เป็นไลบรารี่ขนาดเล็กที่ถูกเรียกใช้ให้ทำงานเฉพาะบางอย่างโดยโปรแกรมหลัก DLL จะถูกโหลดเมื่อจำเป็นต้องใช้งานเท่านั้น ด้วยคุณสมบัตินี้ทำให้ประหยัดหน่วยความจำ RAM (แรม) ได้ โดยไม่ต้องโหลดทั้งโปรแกรมไว้ในหน่วยความจำของเครื่องคอมตลอดเวลา ไฟล์ DLL จะมีนามสกุลเป็น .dll สามารถแบ่งการใช้งานโดยแยก Function การทำงานที่เหมือนกัน ออกมาทำเป็น File DLL เพื่อเรียกใช้งานได้หลายๆครั้ง ทำให้พื้นที่จัดเก็บลดลง ง่ายต่อการเปลี่ยนแปลง เมื่อต้องการเปลี่ยนแปลงสามารถปรับแก้ได้ที่ File DLL เพียงที่เดียวทำให้แก้ไขได้สะดวกการ สามารถนำ Function ใน DLL ไปใช้งานได้ทันทีไม่ต้องพัฒนาใหม่หรือคัดลอกไปใช้ได้

1. ปรับปรุงโปรแกรมตัวอย่างให้เป็น Dynamic-link library พร้อมอธิบาย