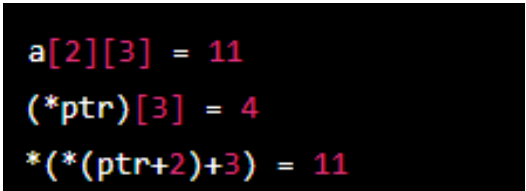


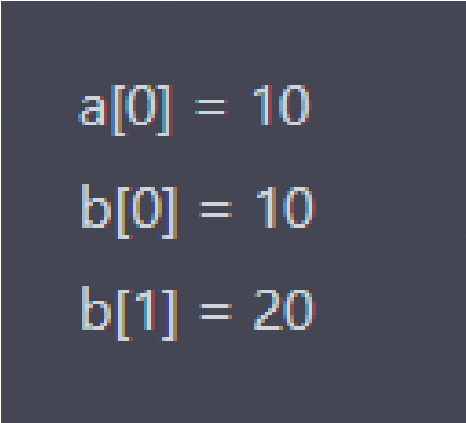
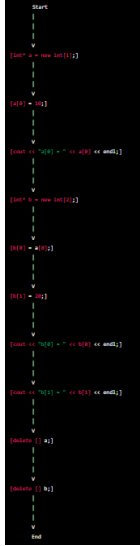
ตอนที่ 1 จงอธิบายความหมายพร้อมยกตัวอย่างประกอบ

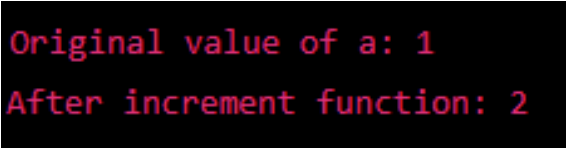
จงเขียนคำอธิบาย ยกตัวอย่างประกอบ และวาดรูปประกอบตามความเข้าใจของคุณ

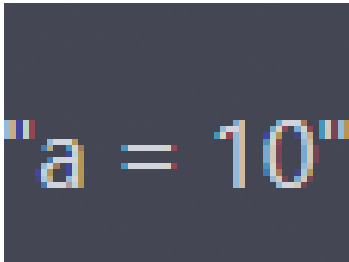
ข้อที่ 1 จงอธิบายความหมายของ Pointer อย่างละเอียด และยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน Pointer	
คำอธิบาย	Code ตัวอย่าง
<p>ในภาษา C คุณสามารถใช้ตัวชี้หน่วยความจำ (pointer) เพื่อชี้ไปยังที่อยู่ของข้อมูลที่อยู่ในหน่วยความจำของโปรแกรม</p>	<pre>int main() { int a = 5; int *ptr = &a; // ptr ชี้ไปที่ตำแหน่งของ a ในหน่วยความจำ printf("a = %d\n", a); // ค่าของ a คือ 5 printf("*ptr = %d\n", *ptr); // ค่าของ *ptr คือค่าของ a ซึ่งคือ 5 printf("ptr = %p\n", ptr); // ที่อยู่ของ ptr คือที่อยู่ของ a ในหน่วยความจำ return 0; }</pre>
<p>ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)</p>	<p>Flow chart ของ Code ตัวอย่าง</p> <pre> graph TD Start([Start]) --> DeclA[Declare int a = 5] DeclA --> DeclPtr[Declare ptr ptr = &a] DeclPtr --> PrintA[printf("a = %d\n", a)] PrintA --> PrintPtr[printf("ptr = %p\n", ptr)] PrintPtr --> PrintPtrVal[printf("ptr = %d\n", *ptr)] PrintPtrVal --> End([End]) </pre>

ข้อที่ 2 จงสร้าง Pointer จำนวน 1 ตัวที่ชี้ Array ไม่จำกัดแถว แถวละ 4 Column และยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน	
คำอธิบาย	Code ตัวอย่าง
<p>ประกาศ Array ชื่อ a ขนาด 5 แถว 4 column ประกาศตัวแปร ptr เป็นพอยเตอร์และกำหนดค่าเป็นที่อยู่ของ Array a ซึ่งคือ Array ชื่อ a แถวละ 4 column แสดงผลค่าของ a[2][3] ซึ่งคือค่าที่อยู่ใน Array a บรรทัดที่ 3 หลักที่ 4 ซึ่งคือ 11</p>	<pre>#include <stdio.h> int main() { int a[5][4] = { {1, 2, 3, 4}, {5, 6, 7, 8}, {9, 10, 11, 12}, {13, 14, 15, 16}, {17, 18, 19, 20} }; int (*ptr)[4] = a; // ptr ชี้ไปที่ Array แถวละ 4 column printf("a[2][3] = %d\n", a[2][3]); // ค่าของ a[2][3] คือ 11 printf("(ptr)[3] = %d\n", (*ptr)[3]); // ค่าของ (ptr)[3] คือค่าของ a[0][3] ซึ่งคือ 4 printf("*(ptr+2)+3 = %d\n", (*(ptr+2)+3)); // ค่าของ *(ptr+2)+3 คือค่าของ a[2][3] ซึ่งคือ 11 return 0; }</pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	Flow chart ของ Code ตัวอย่าง
	

ข้อที่ 3 จงอธิบายเรื่อง Pointer Functionยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน	
คำอธิบาย	Code ตัวอย่าง
<p>Pointer function คือ ฟังก์ชันที่มีการเรียกใช้งานด้วยการใช้พอยเตอร์(pointer) เพื่อเข้าถึงข้อมูลบนหน่วยความจำของเครื่อง สามารถเขียนฟังก์ชันด้วย pointer ได้หลายรูปแบบ ดังนี้:</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p>	<pre>#include <iostream> using namespace std; void changeValue(int* x) { *x = 20; } int main() { int a = 10; cout << "a = " << a << endl; changeValue(&a); cout << "a = " << a << endl; return 0; }</pre> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p> <p>.....</p>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	Flow chart ของ Code ตัวอย่าง
<p>a = 10 a = 20</p>	<pre>graph TD Start([Start]) --> Init([int a = 10]) Init --> Print1([cout << "a = " << a << endl]) Print1 --> CallFunc([changeValue(&a)]) CallFunc --> Print2([cout << "a = " << a << endl]) Print2 --> End([End])</pre>

ข้อที่ 4 จงอธิบายเรื่อง Dynamic Array ยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน	
คำอธิบาย	Code ตัวอย่าง
<p>Dynamic array คือ อาร์เรย์ที่มีขนาดที่เปลี่ยนแปลงได้ตามความต้องการ ซึ่งจะมีการสร้างหน่วยความจำใหม่ขึ้นมาตามความต้องการเพื่อเก็บข้อมูล ดังนี้:</p>	<pre>#include <iostream> using namespace std; int main() { // สร้าง dynamic array ขนาด 1 int* a = new int[1]; a[0] = 10; cout << "a[0] = " << a[0] << endl; // ขยายขนาด dynamic array ขึ้นอีก 1 int* b = new int[2]; b[0] = a[0]; b[1] = 20; cout << "b[0] = " << b[0] << endl; cout << "b[1] = " << b[1] << endl; // ลบหน่วยความจำของ dynamic array delete [] a; delete [] b; return 0; }</pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	Flow chart ของ Code ตัวอย่าง
	 <pre>graph TD Start([Start]) --> A["int* a = new int[1];"] A --> B["a[0] = 10;"] B --> C["cout << \"a[0] = \" << a[0] << endl;"] C --> D["int* b = new int[2];"] D --> E["b[0] = a[0];"] E --> F["b[1] = 20;"] F --> G["cout << \"b[0] = \" << b[0] << endl;"] G --> H["cout << \"b[1] = \" << b[1] << endl;"] H --> I["delete [] a;"] I --> J["delete [] b;"] J --> End([End])</pre>

ข้อที่ 5 จงอธิบายการส่งผ่านตัวแปรแบบ Pass by reference ยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน	
คำอธิบาย	Code ตัวอย่าง
<p>Pass by reference คือการส่งออบเจกต์ (object) ที่มีอ้างอิง (reference) เข้าไปในฟังก์ชัน หรือการส่งค่าอ้างอิง (reference value) เข้าไปในฟังก์ชัน ซึ่งจะทำให้ค่าของออบเจกต์นั้นสามารถถูกเปลี่ยนแปลงในฟังก์ชัน และการเปลี่ยนแปลงจะส่งผลต่อค่าของออบเจกต์นั้นที่อยู่นอกฟังก์ชันด้วย</p>	<pre>#include <stdio.h> // Function that takes a pointer to an integer and increments the value of the integer by 1 void increment(int *x) { (*x)++; } int main(void) { int a = 1; printf("Original value of a: %d\n", a); increment(&a); printf("After increment function: %d\n", a); return 0; }</pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	Flow chart ของ Code ตัวอย่าง
	

ข้อที่ 6 จงอธิบายการส่งผ่านตัวแปรแบบ Pass by value ยกตัวอย่างการใช้งาน พร้อม Code ตัวอย่างการใช้งาน	
คำอธิบาย	Code ตัวอย่าง
<p>การ "pass by value" คือการส่งค่าตัวแปรจากฟังก์ชันหนึ่งไปยังฟังก์ชันอื่นโดยการสำเนาของตัวแปรนั้น และส่งสำเนาของมันไปให้กับฟังก์ชันอื่น โดยฟังก์ชันที่ได้รับสำเนาจะไม่สามารถทําการแก้ไขค่าของตัวแปรต้นฉบับได้</p>	<pre>#include <stdio.h> void add_one(int x) { x = x + 1; } int main() { int a = 10; add_one(a); printf("a = %d\n", a); return 0; }</pre>
ผลลัพธ์ของ Code (Captureพร้อมแปะรูป)	Flow chart ของ Code ตัวอย่าง
	<pre> +-----+ start +-----+ v +-----+ int a = 10 +-----+ v +-----+ add_one +-----+ v +-----+ print "a = %d" +-----+ v +-----+ end +-----+ </pre>