



มหาวิทยาลัยอัสสัมชัญ
คณะวิศวกรรมศาสตร์

สอบกลางภาค: ภาคการศึกษาที่ 1

ปีการศึกษา: 2547

วันที่สอบ: 3 สิงหาคม 2547

เวลาสอบ: 13.30 – 16.30 น.

รหัสวิชา: 240-101

ห้องสอบ: A201, A203, A205, A301,
A303, A305, A401

ชื่อวิชา: Introduction to Computer Programming

คำสั่ง: อ่านรายละเอียดของข้อสอบ และคำแนะนำให้เข้าใจก่อนเริ่มทำข้อสอบ

อนุญาต: เครื่องเขียนต่างๆ เช่น ปากกา หรือดินสอ

ไม่อนุญาต: หนังสือ, เอกสารใดๆ และเครื่องคิดเลข

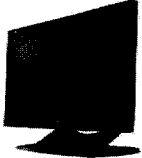

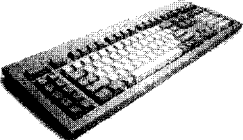
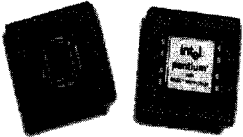

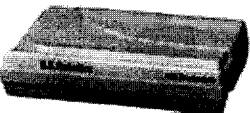

เวลา: 3 ชั่วโมง (180 นาที)

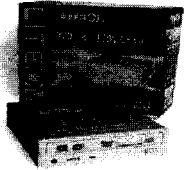


คำแนะนำ

- ข้อสอบมี 17 หน้า (รวมใบปะหน้า) แบ่งเป็น 5 ตอน คะแนนรวม 100 คะแนน
- คำตอบทั้งหมดจะต้องเขียนลงในข้อสอบ รวมทั้งเขียนชื่อและรหัสให้ชัดเจนทุกแผ่น ถ้าแผ่นใดของข้อสอบไม่มีชื่อและรหัส นักศึกษาจะถูกหักคะแนน 0.5 คะแนนต่อแผ่น
- คำตอบส่วนใดอ่านไม่ออก จะถือว่าคำตอบนั้นผิด
- การเขียนโปรแกรม ให้ใช้ภาษา C++
- อ่านคำสั่งในแต่ละข้อให้เข้าใจก่อนลงมือทำ
- หากข้อใดเขียนตอบไม่พอ ให้เขียนเพิ่มเติมด้านหลังของหน้านั้นเท่านั้น
- **ทุจริตในการสอบ** โทษขั้นต่ำคือปรับตกในรายวิชานี้และพักการเรียนหนึ่งภาคการศึกษา โทษสูงสุดคือไล่ออก

ตอนที่ 1 (20 คะแนน)

1 จงระบุชื่ออุปกรณ์ในรูป พร้อมทั้งอธิบายว่าอุปกรณ์ดังกล่าว ทำหน้าที่อย่างไร ในระบบคอมพิวเตอร์ (หน่วยประมวลผล, อุปกรณ์อินพุต, อุปกรณ์เอาต์พุต, หน่วยความจำ) (10 คะแนน)

ข้อ	รูป	ชื่อ	หน้าที่
1.1			
1.2			
1.3			
1.4			
1.5			
1.6			
1.7			

ข้อ	รูป	ชื่อ	หน้าที่
1.8			
1.9			
1.10			

2. เครื่อง Unicom ใช้ระบบปฏิบัติการอะไร การเข้าไปใช้งานเครื่อง Unicom ทำได้โดยการใช้บริการอะไรในระบบอินเทอร์เน็ต (2 คะแนน)

.....

.....

3. ต้องการย้ายไดเรกทอรี /home/s4710555/homework พร้อมไฟล์ทั้งหมด ที่อยู่ในนี้ไปยัง /home/s4710666 ทำได้อย่างไร จงแสดงคำสั่ง (3 คะแนน)

.....

.....

4. introcom@coe.psu.ac.th คืออะไร มีความสำคัญอย่างไรในระบบอินเทอร์เน็ต (2 คะแนน)

.....

.....

5. 203.146.155.20 และ www.coebiz.net เป็นข้อมูลที่ได้จากเครื่องคอมพิวเตอร์เครื่องเดียวกัน จงอธิบายว่า ข้อมูลทั้งสองอย่าง มีความสัมพันธ์กันอย่างไร (3 คะแนน)

.....

.....

/** จบตอนที่ 1 อย่าลืมเขียนชื่อ รหัส และ Section ลงในข้อสอบทุกแผ่น**/

ตอนที่ 2 (25 คะแนน)

1. จงใส่เครื่องหมาย ✓ หน้าข้อความที่ถูกต้อง และใส่เครื่องหมาย X หน้าข้อความที่ไม่ถูกต้อง พร้อมให้เหตุผลว่าทำไมถึงไม่ถูกต้อง (7คะแนน)

..... 1.1 ในการประกาศตัวแปร จะต้องระบุชนิดของตัวแปรและค่าเริ่มต้นของตัวแปรเสมอ

เหตุผล

..... 1.2 ถ้าต้องการให้โปรแกรมพิมพ์ข้อความ 3 บรรทัดออกทางหน้าจอ สามารถใช้คำสั่ง cout เพียงครั้งเดียวได้

เหตุผล

..... 1.3 ไฟล์ที่มีนามสกุล .obj เป็นไฟล์ที่สามารถทำงาน (execute) ได้

เหตุผล

..... 1.4 1a_2b สามารถนำมาตั้งเป็นชื่อตัวแปรใน c++ ได้

เหตุผล

..... 1.5 คอมเมนต์ หรือหมายเหตุ เป็นส่วนที่ทำให้คอมไพเตอร์พิมพ์ข้อความหลังเครื่องหมาย // ออกทางหน้าจอ เมื่อทำการรันโปรแกรม

เหตุผล

..... 1.6 ตัวดำเนินการ % เป็นตัวดำเนินการที่กำหนดให้ใช้กับค่าจำนวนเต็มเท่านั้น

เหตุผล

..... 1.7 a= b; เป็นคำสั่งที่ใช้ในการเปรียบเทียบว่าค่าในตัวแปร a เท่ากับค่าในตัวแปร b หรือไม่

เหตุผล

2. จงเลือกตัวอักษรทางด้านขวามาเติมในช่องว่างด้านซ้ายมือ โดยเนื้อหาทั้งคู่จะต้องมีความสัมพันธ์กันมากที่สุด (5 คะแนน)

..... /*...*/

a. char

..... if

b. string

..... g++

c. linker

..... "I am happy"

d. ASCII

..... #include

e. return 0;

f. comment

g. variable

h. compiler

i. keyword

3. จงเขียนผลลัพธ์ของคำสั่งต่อไปนี้ โดยใช้ตารางลำดับการทำงานของตัวดำเนินการข้างล่าง เมื่อกำหนดตัวแปร a และ b ในแต่ละข้อ ให้เป็นจำนวนเต็ม และ a=2 และ b=3 (5 คะแนน)

- 3.1 cout<< (a += --b); ผลลัพธ์.....
- 3.2 cout<< a*b+1/10*b-8; ผลลัพธ์.....
- 3.3 cout<< "a"<< (a--b-2); ผลลัพธ์.....
- 3.4 cout<< a/b<<"!="<< a%b; ผลลัพธ์.....
- 3.5 cout<< (!a<b&&b); ผลลัพธ์.....

4. จากส่วนหนึ่งของโปรแกรม จงเขียนผลลัพธ์ที่แสดงออกทางหน้าจอคอมพิวเตอร์ (3 คะแนน)

```
int x=1, y=3;
float z;
cout<<"\n"+x<<"\n"<<endl;
z=(float)x/y;
cout<<"\n"<<z<<"\n";
cout<<setprecision(3)<<z<<endl;
```

.....

.....

.....

.....

.....

ลำดับการทำงานก่อน-หลังของตัวดำเนินการเป็นไปตามตารางข้างล่างนี้

ตัวดำเนินการ	ทิศทางการดำเนินการ
() , []	ซ้ายไปขวา
!, ++, --, +(ค่าบวก), -(ค่าลบ), sizeof	ขวาไปซ้าย
*, /, %	ซ้ายไปขวา
+, - (ตัวกระทำทางคณิตศาสตร์)	ซ้ายไปขวา
<, <=, >, >=	ซ้ายไปขวา
==, !=	ซ้ายไปขวา
&&	ซ้ายไปขวา
	ซ้ายไปขวา
=, +=, -=, /=, %=, !=, <<=, >>=	ขวาไปซ้าย

ตอนที่ 3 (20 คะแนน)

1. จากส่วนหนึ่งของโปรแกรมต่อไปนี้ จงหาค่าผลลัพธ์หรือเติมส่วนของผลลัพธ์ที่แสดงออกทางหน้าจอ คอมพิวเตอร์ให้สมบูรณ์ โดยกำหนดให้ข้อความที่เป็นตัวหนาคือค่าที่รับจากผู้ใช้ (14 คะแนน)

ข้อ	ส่วนหนึ่งของโปรแกรม	ผลลัพธ์ที่แสดงออกทางหน้าจอ
1.1	<pre>int x = -1; if (x > 0) cout << 1 << endl; else if (x == 0) cout << 2 << endl; else cout << 3 << endl;</pre>	
1.2	<pre>int op = 3; switch(op){ case 1: cout << '+' << endl; break; case 2: cout << '-' << endl; break; case 3: cout << '*' << endl; break; case 4: cout << '/' << endl; break; default: cout << "Invalid" << endl; }</pre>	
1.3	<pre>int a = 7, num = 0; while (a > 0){ num++; a = a - 2; } cout << num << endl;</pre>	
1.4	<pre>int time = 0; for (int b = 3; b <= 14; b++){ time = time + 1; } cout << time << endl;</pre>	
1.5	<pre>int row = 4; int col = 3; for (int i = 0; i < row; i++){ for(int j = 0; j < col; j++){ if (j >= col/2) cout << 'x'; else cout << 'o'; } cout << endl; }</pre>	

1.6	<pre>char cmd; do { cout << "Enter command: "; cin >> cmd; switch(cmd){ case 'o': cout << "Open\n" ; break; case 's': case 'S': cout << "Search\n"; break; case 'w': cout << "Write\n"; break; case 'x': break; default: cout <<"Invalid command\n"; } }while(cmd != 'x'); cout << "Bye" << endl;</pre>	<p>Enter command: s</p> <p>Enter command: x</p>
-----	---	---

2. จงเติมส่วนหนึ่งของโปรแกรมต่อไปนี้เพื่อให้ทำงานได้ตามรายละเอียดที่กำหนด (6 คะแนน)

ข้อ	ส่วนหนึ่งของโปรแกรม	รายละเอียด
2.1	<pre>int count = 0; for(int k = 1; k < 100; k++){ if ((k%3 == 0) _____) count++; } cout << "# of integers divisible by 3 or 5 is "; cout << count << endl;</pre>	<p>ตัวแปร count เก็บจำนวนของจำนวนเต็มระหว่าง 1 ถึง 100 ที่หารด้วย 3 หรือ 5 ลงตัว</p> <p>ตัวอย่างของผลลัพธ์คือ</p> <p># of integers divisible by 3 or 5 is 46</p>
2.2	<pre>int input; cout<<"Enter a positive integer: "; cin >> input; cout << "Its reverse is "; while(_____) { cout << input%10; input = input/10; } cout << endl;</pre>	<p>รับค่าจำนวนเต็มจากผู้ใช้แล้ว</p> <p>แสดงตัวเลขแต่ละตัวของจำนวนเต็มนั้นในทิศทางตรงข้าม</p> <p>ตัวอย่างของผลลัพธ์คือ</p> <p>Enter a positive integer: 123</p> <p>Its reverse is 321</p>
2.3	<pre>int total = 0; for(int m = -1; m > -50; _____) total += m; cout << "The sum is " cout << total << endl;</pre>	<p>ตัวแปร total เก็บผลรวมของอนุกรม -1, -4, -7, ..., -49</p> <p>ตัวอย่างของผลลัพธ์คือ</p> <p>The sum is -425</p>

/**/ จบตอนที่ 3 อย่าลืมเขียนชื่อ รหัส และ Section ลงในข้อสอบทุกแผ่น**/

ตอนที่ 4 (20 คะแนน)

1. จงเขียนต้นแบบของฟังก์ชัน (function prototype) ต่อไปนี้ (3 คะแนน)

1.1 ฟังก์ชันชื่อ Sort_data รับพารามิเตอร์เป็นเลขจำนวนเต็ม 3 ค่า ไม่ส่งค่ากลับ

1.2 ฟังก์ชันชื่อ Cal_volume รับพารามิเตอร์เป็นเลขจำนวนเต็ม 3 ค่า ส่งค่ากลับเป็นชนิดจำนวนเต็ม

1.3 ฟังก์ชันชื่อ Check_val รับพารามิเตอร์เป็นอักขระ(character) 1 ค่า ไม่ส่งค่ากลับ

2. จงแสดงการเรียกใช้ฟังก์ชันซึ่งมีต้นแบบของฟังก์ชันดังนี้ (2 คะแนน)

```
char Cal_grade(int score);           // ฟังก์ชัน Cal_grade ส่งค่าระดับเกรด  
char Compare(int score1, int score2); // ฟังก์ชัน Compare ส่งค่า  
อักขระ(character) มากกว่าหรือน้อยกว่า (>,<) ที่ได้จากการเปรียบเทียบคะแนน 2 ค่า
```

2.1 แสดงการเรียกใช้ฟังก์ชัน Cal_grade เพื่อหาว่าคะแนนเท่ากับ 75 อยู่ในระดับเกรดใด โดยเก็บค่าของเกรดที่ได้ไว้ในตัวแปรชื่อว่า grade

2.2 เขียนคำสั่งให้แสดงผลลัพธ์ออกทางหน้าจอว่า

```
score1 > score2  
หรือ  
score1 < score2 ก็ได้ โดยในคำสั่งจะต้องมีการเรียกใช้ฟังก์ชัน Compare
```

3. จงเพิ่มเติมส่วนของโปรแกรมต่อไปนี้ให้สามารถคอมไพล์โปรแกรมผ่าน และทำงานได้อย่างถูกต้อง โดยโปรแกรมมีการทำงาน 2 ส่วนดังนี้

- คำนวณหาความยาวของเส้นรอบวงของวงกลม
- คำนวณหาพื้นที่ของสามเหลี่ยม

หมายเหตุ ให้เพิ่มเติมส่วนของโปรแกรมที่คิดว่าขาดหายไปในช่วงที่กำหนดไว้ โดยไม่จำเป็นต้องเติมทุกช่อง (5 คะแนน)

```
1 #include<iostream.h>
2 #define Pi 3.1416
3 float Circle_perimeter(float);
4 _____
5 int main()
6 {
7     float r,perimeter;
8     _____
9     cout << "Enter radius of circle: ";
10    cin >> r;
11    cout << "Enter width of triangle: ";
12    cin >> width;
13    cout << "Enter height of triangle: ";
14    cin >> height;
15    _____
16    perimeter = _____
17    area = _____
18    _____
19    cout << "Perimeter of circle at radius = " << r <<
" is " << perimeter << endl;
20    cout << "Area of triangle at width = " << width <<
" and height = " << height << " is " << area << endl;
21    return 0;
22 }
23 _____
24 _____
25 {
26     return 2*Pi*R;
27 }
28
29 float Triangle_area(float W,float H)
30 {
31     return 0.5*W*H;
32 }
```

ผลลัพธ์ของการทำงานของโปรแกรมเป็นดังนี้

```
Enter radius of circle:2
Enter width of triangle:3
Enter height of triangle:4
Perimeter of circle at radius = 2 is 12.5664
Area of triangle at width = 3 and height = 4 is 6
```


ตอนที่ 5 (15 คะแนน)

ส่วนของโปรแกรมที่กำหนดให้เป็นโปรแกรมเกมสชื่อ "Twenty one pickup" ซึ่งมีวิธีเล่นดังนี้
เกมสนี้มีผู้เล่น 2 คนคือ player กับ คอมพิวเตอร์ โดยจะมีกองหินซึ่งมีหินอยู่ 21 ก้อน ให้แต่ละผู้เล่น
ผลัดกันหยิบหินคนละครั้ง โดยมีกติกาว่าการหยิบหินแต่ละครั้งผู้เล่นจะต้องหยิบหินอย่างน้อย 1 ก้อนแต่
ไม่เกิน 3 ก้อน ใครเป็นคนหยิบหินคนสุดท้ายคนนั้นจะเป็นผู้ชนะ

ให้นักศึกษาเติมส่วนของโปรแกรมลงในที่ว่าง(1-6)ที่เว้นไว้ให้เพื่อทำให้โปรแกรมสมบูรณ์(เมื่อรันโปรแกรม
แล้วได้ผลลัพธ์(out put) ของโปรแกรมเหมือนผลลัพธ์ที่ให้มา) กำหนดการทำงานของฟังก์ชันให้ดังนี้

1. ฟังก์ชัน `print_instruction` จะเอาไว้แสดงคำอธิบายวิธีเล่นเกมส
2. ฟังก์ชัน `computer_move` จะรับค่าจำนวนก้อนหินที่มีอยู่ในกอง(`number_of_stones`) และทำ
การหาค่าจำนวนก้อนหินที่คอมพิวเตอร์จะหยิบออก(กำหนดให้ใช้ `modulus(%)` ในการคำนวณ)
และส่งค่าจำนวนก้อนหินที่เหลืออยู่ในกองกลับ(หักจำนวนก้อนหินที่ถูกหยิบโดยคอมพิวเตอร์
จากจำนวนก้อนหินที่รับค่ามา)
3. ฟังก์ชัน `get_user_move` จะรับค่าจำนวนก้อนหินที่มีอยู่ในกอง(`number_of_stones`)
และถาม `player` ว่าต้องการหยิบก้อนหินออกกี่ก้อน หลังจากรับค่าที่ `player` พิมพ์ค่า ให้ตรวจ
สอบด้วยว่าค่าที่ได้รับมาเป็นไปตามกติกาหรือไม่ ถ้าไม่ก็ให้แสดงข้อความ(error message)บอก
`player` และให้ `player` ใส่ค่าใหม่จนกว่าค่าที่ได้รับจะถูกต้อง แล้วจึงส่งค่าจำนวนก้อนหินที่ถูก
หยิบออก

โปรแกรม Twenty_one_pickup.cpp

```
//twenty_one_pickup.cpp
#include<iostream.h>
void print_instruction(void);
int  player_move(int number_of_stones);
int  computer_move(int number_of_stones);
int  get_user_move(int number_of_stones);

int main(void){
    int number_of_stones;//จำนวนก้อนหิน
    bool player_moved_last;//เอาไว้ตรวจสอบว่าใครเป็นคนสุดท้ายที่หยิบก้อนหิน
    number_of_stones=21;//กำหนดให้มีก้อนหิน 21 ก้อน
    player_moved_last=false;

    print_instruction();
    while(_____(1)_____){
        number_of_stones=player_move(number_of_stones);
        player_moved_last=true;
        if(number_of_stones>0){
            number_of_stones=computer_move(number_of_stones);
            player_moved_last=false;
        }
    }
    //print the outcome
    if(player_moved_last)
        cout<<"Congratulations, you won."<<endl;//player ชนะ
    else
        cout<<"You lose, try again."<<endl; //computer ชนะ
}
void print_instruction(void){
    cout<<"The object of this game is to remove the last stone."
<<endl;
    cout<<"There are 21 stones in the pile to start with."<<endl;
    cout<<"You may remove from 1 to 3 stones on each move."<<endl;
}
}
```


The computer removes 2 stone(s) leaving 16 stone

Your move - how many stones do you wish to remove?

2

There are 14 stones remaining.

The computer removes 2 stone(s) leaving 12 stone

Your move - how many stones do you wish to remove?

1

There are 11 stones remaining.

The computer removes 3 stone(s) leaving 8 stone

Your move - how many stones do you wish to remove?

3

There are 5 stones remaining.

The computer removes 1 stone(s) leaving 4 stone

Your move - how many stones do you wish to remove?

1

There are 3 stones remaining.

The computer removes 3 stone(s) leaving 0 stone

you lose, try again.

/* จบตอนที่ 5 อย่าลืมเขียนชื่อ รหัส และ Section ลงในข้อสอบทุกแผ่น***/**