



มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี  
ข้อสอบกลางภาคการศึกษาที่ 2/2558

วันศุกร์ที่ 26 กุมภาพันธ์ 2559

เวลา 09.00 - 12.00 น.

วิชา CPE 113 Algorithms and Data Structures.

น.ศ. วศ.คอมพิวเตอร์ชั้นปีที่ 1A,B กลุ่มที่ 1,2

คำสั่ง

1. ข้อสอบมีทั้งสิ้น 7 ข้อ จำนวน 8 แผ่น(รวมแผ่นนี้) คิดเป็นคะแนนรวมทั้งสิ้น 35 คะแนน
2. ทำทุกข้อลงในข้อสอบที่เว้นช่องไว้ให้
3. ไม่อนุญาตให้นำเครื่องคำนวณใดๆเข้าห้องสอบ
4. ห้ามนำเอกสารใดๆ เข้าห้องสอบ
5. เขียนชื่อ และ รหัสประจำตัว ลงในกระดาษคำตอบทุกแผ่น

(อ.พิพัฒน์ ศุภศิริสันต์ )

ผู้ออกข้อสอบ

024709082

ข้อสอบนี้ได้ผ่านการประเมินจากภาควิชาวิศวกรรมคอมพิวเตอร์แล้ว

ชื่อ .....รหัสประจำตัว.....ภาควิชา/ชั้นปี.....

1. ในการแปลงเลขฐาน 10 ให้เป็นเลขฐานใดๆ โดยวิธีเรียกตัวเอง มีตัวอย่างการคิดหาคำตอบดังนี้  
ต้องการแปลง 25 ฐานสิบ ให้เป็นเลขฐาน 2 จะได้

คำตอบของ(25)  $(25/2 = 12 \text{ เศษ } 1)$  คำตอบคือ  $1 + 10^{\text{คำตอบของ}(12)} = 1 + 10^{1100} = 11001$

คำตอบของ(12)  $(12/2 = 6 \text{ เศษ } 0)$  คำตอบคือ  $0 + 10^{\text{คำตอบของ}(6)} = 0 + 10^{110} = 1100$

คำตอบของ(6)  $(6/2 = 3 \text{ เศษ } 0)$  คำตอบคือ  $0 + 10^{\text{คำตอบของ}(3)} = 0 + 10^{11} = 110$

คำตอบของ(3)  $(3/2 = 1 \text{ เศษ } 1)$  คำตอบคือ  $1 + 10^{\text{คำตอบของ}(1)} = 1 + 10^1 = 11$

คำตอบของ(1)  $(1/2 = 0 \text{ เศษ } 1)$  คำตอบคือ  $1 + 10^{\text{คำตอบของ}(0)} = 1 + 10^0 = 1$

คำตอบของ(0) = 0

จงสร้างฟังก์ชันที่ทำงานแบบ recursion ที่สอดคล้องกับการเรียกใช้ใน main เพื่อ return คำตอบที่ต้องการกำหนดให้มีการเขียนฟังก์ชัน main() ไว้ดังนี้

```
int main()
{ int num, base;
  long long answer ;
  printf("Program Convert a Number Decimal System to Base N System\n");
  printf("Enter a decimal number : ");
  scanf("%d", &num);
  printf("Enter new base number : ");
  scanf("%d", &base);
  answer = convert_base(dec, base);
  printf("The base %d equivalent of %d is %llu\n", base, num, answer);
  return 0;
}
```

หมายเหตุ ให้ระวังชนิดข้อมูลที่จะส่งกลับด้วย

.....

.....

.....

.....

.....

.....

.....

.....

.....



ชื่อ ..... รหัสประจำตัว.....เลขที่นั่งสอบ.....

3. จงสร้างฟังก์ชันโดยใช้เทคนิค Insertion Sort เพื่อเรียงลำดับข้อมูลที่อยู่ในตัวแปร dict[] โดยใช้ฟิลด์ vocab เป็นคีย์ในการจัดเรียง

**void insertion\_sort (dict\_type dict[], int count)**

{

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

}







พิจารณาตัวอย่างการเขียนฟังก์ชันในการแปลงอาร์เรย์ token[] ซึ่งเก็บ token ของสมการ infix ให้อยู่ในรูปอาร์เรย์ของสมการ postfix ที่กำหนดให้บางส่วน แล้วเขียนคำสั่งเพิ่มเติมเฉพาะกรณีที่ตรวจสอบ token ว่าเป็น operator "(" และ ")" เท่านั้น

**void change\_token\_to\_postfix(char token[][20],int count)**

**{ int i, group;**

**char buff[20];**

**double num;**

**for (i=0; i<count; i++)**

**{ group = check\_group(token[i]);**

**if (group==1) // number**

**add\_postfix(token[i]);**

**else if (group>=2&&group<=4) // operator**

**replace\_operator\_stack (token[i]);**

**else if (group==5) // open parenthesis**

.....  
 .....  
 .....  
 .....  
 .....  
 .....

**else if (group==6) // close parenthesis**

.....  
 .....  
 .....  
 .....  
 .....  
 .....  
 .....  
 .....  
 .....

**}**

**}**