Name,Surname:..:....................   CMPE108 Group: .... Signature............................

Please print this homework, and solve all questions on the printout using a pen or pencil. Do not forget to write your student number on every page. Late submissions, printout solutions, and photocopies will not be graded.

The following information will be available on midterm exam sheets.

Precedence and Associativity Table

Operators	Associativity
( ) [ ] -> .	Left to right
! ++ -- + – * & (type)	Right to left (Unary)
* / %	Left to right
+ –	Left to right
< <= > >=	Left to right
== !=	Left to right
&&	Left to right
||	Left to right
?:	Right to left (ternary)
= += -= *= /= %=	Right to left
,	Left to right

% specifiers in ANSI C:

%c char single character   **%o** int unsigned octal value
%d (**%i**) int signed integer   **%p** pointer address stored in pointer
%e (**%E**) float or double exponential format   **%s** array of char sequence of characters
%f float or double signed decimal   **%u** int unsigned decimal
%g (**%G**) float or double use **%f** or **%e** as required   **%x** (**%X**) int unsigned hex value

String Control Codes:

\a alert,   **\b** backspace,   **\f** formfeed,   **\n** new line,   **\r** carriage return,
\t horizontal tab,   **\'** single quote,   **\0** null.   **** backslash.

Q1.

a) From the following items encircle the C keywords. (Each false mark cancels one correct mark!)

 if,   **while**,   **int**,   **include**,   **stdio**,   **for**,   **Pause**,   **double**,   **break**,   **stop**,   **end**

b) State which of the following C identifier names are valid or invalid. For the invalid names, propose a similar but valid name.

i) **Cmpe_108**   _____

ii) **dollar$**   _____

iii) **printf_scanf**   _____

iv) **747_Boeing**   _____

v) **football club**   _____

vi) **Name**   _____

vii) **Name Surname**   _____

viii) **20number**   _____

ix) **number#**   _____

x) **double**   _____

xi) **_age**   _____

Q2. a) For the following statements, give the corresponding outputs into the boxes on the right which correspond to different spaces in the output.

i) **printf("k=%3d i=%4d", 5, 12);**

ii) **printf("%4.3f", 1410.22);**

iii) **printf("x=%8.1f% -3dm", 0.55, 55);**

b) Write the characters (each character in one box) printed by the following code section.

```
int j=17, k=7, p=1;
float m=33.66, n=3.0;
printf("abc%3de%.1fg\nh%d", j%k, m/n, p++);
```

c) What is the value of i, j, k, f, g after entering the following line for the scanf sentence:

```
int  i,j,k; float f, g;
scanf("%d%d%d%f%f",&i,&k,&j,&g,&f);
```

| 1 | 1 | | 2 | | 3 | 3 | 3 | | 4 | . | 4 | | 5 | . | 5 | |

i= ....   j=.................... k=................   f=................   g=...................

Q3 a) Find the value of the following expressions:

Expression   Result

```
i)   20.0 – 2 / 6 + 3
```
....
```
ii)  10 + 17 % 3 + 4.0
```
....
```
iii) (10 + 3 ) % 4
```
....
```
iv)  3 % 2 / 5
```
....
```
v)   2 * 1 - 2.0 / 4
```
....

b) Given the following declarations:

```
int i=5, j=7, k=12, z=0, a=5, b=2, c=4, d=6;
char ch2= 'b', ch3='B';
```

Evaluate each of the following expressions and write the answer in the box provided

i)   `i + 2 && k -1`

ii)   `i + 2 * j == k`

iii)   `k / 3 != - j + 3 * - i`

iv)   `ch2 != ch3 = = z !== a`

v)   `d % b == c % b || ch2 < ch3`

vi)   `a – b * c || c / b && a`

c) Evaluate the following expressions for the declared values of the variables, and fill its final type (integer or float) and numerical value into answering box.

```
i) int  i=2, n= 15;
   float  x=2.4;
   x*2 + n/i  % 3                type:              value:
```

```
ii) int  m=5;
    float y=4.6;
    m = y + m /                 type:              value:
```

iii) `int    m=5;`
   `2.0f * m / 2`   type: value:

iv) `int    x=5;`
   `float i=4.5;`
   `i + ++ x - -- x`   type: value:

v) `int n=-5;`
   `n += 1 / 2+3 *4% 5-6 x`   type: value:

d) Write the C expression of each real numbered mathematical expressions, without using any unnecessary parenthesis. Do not forget to add a decimal point to the integers to convert them to real numbers.

i) $\dfrac{Q}{1+\dfrac{Q}{R+1}}$ ....

ii) $\dfrac{\dfrac{5}{4}+2i}{3j-\dfrac{4}{5}k}$ ....

iii) $\dfrac{x}{y}-\dfrac{\dfrac{x}{y}+2}{3z^2}$ ....

iv) $2z\dfrac{2x}{x+y}$ ....

v) $\dfrac{x}{y}-x^2+y^2$ ....

vi) $\dfrac{1.0+a}{1-\dfrac{c}{1+ac}-c}$ ....

vii) $c+\dfrac{1/b+c}{b/c+1}$ ....

viii) $\dfrac{(a+b)}{(c\times d)-(e/f)}g^3$ ....

ix) $\dfrac{1}{2}x^2+\dfrac{xy}{x+\frac{1}{x}}$ ....

x) $\dfrac{\frac{2}{3}(x^2)}{\frac{1}{4}+x}$ ....

e) Write down a mathematical expression to each of the following expressions without parenthesis.

   i) `sqrt((b*b) – (4*a*c))` ....

   ii) `2*n*n*n – 4*n*n + 3*n + 1` ....

   iii) `(a*a – b*b)/(a – b) + 1/a + 1/b` ....

f) Considering the precedence of operators put a cross mark into the box of unnecessary pair-of-parenthesis. Example: `( a+b ) * ( c/d ) – (3/ ( 2*a ) )`
                                  [ ]   [ ]   [x]   [x]   [x]   [ ]   [ ] [x]

i)   `( a*b ) / (c/d)*( ( a-b) – c)`
        [ ]   [ ]   [ ] [ ][ ] [ ]   [ ]   [ ]

ii)   `( a*b ) + (c/d)*( a-b )`
        [ ]   [ ]   [ ] [ ][ ]   [ ]

iii)   `( (5./4.) + (2.*i) )/ ( (3.*j) – (4./5.)*k )`
        [ ][ ]   [ ]   [ ] [ ][ ] [ ][ ]   [ ]   [ ] [ ]

3

g) For the following display outputs fill in the blanks of the printf statements to get the output exacly as given below. Note that in codes and output "_" represents the space character.

i. `printf("..........................", 256)`   to get the output: **x=_ _ 256m**

ii. `printf("..........................", -92.39538)`   to get the output: **h:-92.40_ _ _ _ m**

iii. `printf("........................", 78.461032)`   to get the output: **_ _ _ 78.46m**

iv. `printf("......................", 1.2345678)`   to get the output: **_ _ 1.2e000m**

v. `printf(".......%..............", 5)`   to get the output: **t=_ _ 005 min**

vi. `printf("%...................", 40)`   to get the output: **40_ _ _ _ _ _ m**

vii. `printf("%..................", 3.26)`   to get the output: **_ 3.3**

viii. `printf("%.................", 100)`   to get the output: **_ _ 0100**

ix. `printf("%...................", 55.33)`   to get the output: **55.3300_ _ _Nt**

x. `printf("%...................", 43)`   to get the output: **_ _ _ 43**

xi. `printf("%...................", 7)`   to get the output: **_ _ 007**

h) Fill in the blanks to read two integers and one floating point numbers (**i**, **j**, and **a**) correctly.

```
int i, j; float a;

scanf(.....................................................................);
```

Q4

The following program calculates workers' salary. (work_code and work_hour is entered by user.)

Find the output of the following program for three different sets of inputs as given below:

a) `work_code`: 2   b) `work_code`: 3   c) `work_code`: 1

   `work_hour`: 25   `work_hour`: 10   `work_hour`: 30

Find the output and write your answer in the corresponding answer box.

```c
#include <stdio.h>
#include <stdlib.h>
int main (void)
{    int work_code, work_hour;
     double salary;
     printf("code?: ");
     scanf("%d",&work_code);
     if ((work_code ==2)||(work_code ==3))
     {    printf("work hour?: ");
          scanf("%d",&work_hour);
     }
     switch(work_code)
     {  case 1: salary=875;
              break;
        case 2: salary=6.80*work_hour;
              break;
        case 3: salary=12.5*(work_hour%40)+875;
     }
     printf("salary: %.2f\n",salary);
     system("pause");
     return (0);
}
```

a)

b)

c)

4

Q5. a) The following C program contains 12 errors. Find each error and show it by putting a circle around it. (i.e. circle only the exact location of the error, you do not need to correct the errors, you will lose -1 point for each wrongly identified error)

```
#include   stdio.h>
/* This program finds your age
#define   YearOfCalendar = 2017
int main (void)
{
    int  birthYear  age
    printf(    "Enter Your Birth Year, in four digits );
    scanf ( "%d",  birthyear);
    printf("We are in the year %f, and ", YearofCalendar );
    age=YearOfCalendar - birthYear;
    printf ("your age is calculated as %d", &age );
    return0;
}
```

b) The following C program contains some programming mistakes in five statements. Find the lines with programming mistake(s) and write down the corrected statements in the boxes given below:

```
1.    // This program computes the sum of digits of
2.    // a 3-digit integer number:
3.    // (e.g. If input = 234, output = 9)
4.    #include <standardio.h>
5.    int main(void) {
6.      int num, a, b, d, SUM;
7.      printf("Enter a 3-digit number : ");
8.      scanf("%d", num);
9.      a = num % 10;
10.     num= num/10
11.     b = num % 10;
12.     num= num/10;
13.     d = num;
14.     Sum  = a+b+D;
15.     printf ("The num and SUM are = %d \n", num,SUM);
16.     return(0);
17.   }
```

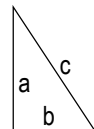| Line number	Corrected statement

Q6.

a) The following C program calculates the travel time of a car running at constant speed for a given distance. Program reads the speed "**kmPerHr**", and the distance "**kmeters**" from the keyboard as integer numbers, calculates time "**minutes**", find hours "**hrs**", and remaining minutes "**minr**". Then, it writes the time in "hours:minutes" format as "**Travel time**=h:m". Complete the missing parts.

Example: if speed is **100**, and distance is **150**, then minutes shall be 60*150/100= 90, hrs shall be 1, minr shall be 30, and display shall be "**Travel time=1:30**".

```
#include <stdio.h>
int main()
{
int kmPerHr, _____;
int _____, hrs, minr ;
/*read speed and distance from the keyboard*/
_____;
/*compute minutes */
_____;
/* divide minutes by 60 to get hrs */
_____;
/* find remainder by 60 to get minr*/
_____;
/* Print the result in hr:min format */
_____;
return 0;
 }
```

b) The following C PROGRAM computes $c = \sqrt{a^2 + b^2}$, the hypotenuse (the side opposite the right angle) of a right triangle, based on the Pythagorean Theorem. There are 10 blanks in the program. Complete each of the blanks.

```
#include <stdio.h>
#include <math.h> /* for the function sqrt */

............. main(void)
{
    float a, b, c;          /* declare floating point variables */

    ............... sq_a, ...............; /* declare floating point variables */

    printf("Insert length of first side \'a\':");

    ...........................................; /* reads 'a' from the input */

    printf("Insert length of second side \'b\':");

    ...........................................; /* reads 'b' from the input */

    ...........................................; /* computes square of 'a' and stores it in 'sq_a' */

    ...........................................; /* computes square of 'b' and stores it in 'sq_b' */

    c = sqrt(sq_a + sq_b); /* sqrt computes the square root of 'sq_a' + 'sq_b'*/

    printf("The length of the hypotenuse \'c\' is ...............", ...............);

    ...........................................; /* terminates the main function normally */
}
```

c) Given the following C program, trace it and write down the **output** of the **printf** statement.

```c
#include<stdio.h>
void main()
{
  int num1 = 100, num2=200, num3;
  num3 = num2/num1 <= ( num1 >= 100 ) + (num2 < 50 );
  printf("%d",num3);
  return 0;
}
```
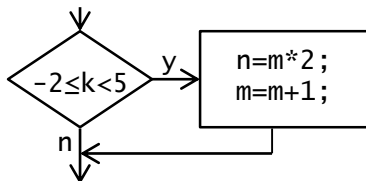
                                output: ……………………………………………………………

Q7)

a) Assuming that **int i, k, n;** already declared write the following C code using switch-case statement

```c
if(k==5) ++n;
else if(k==0)   --n;
else n=0;
```
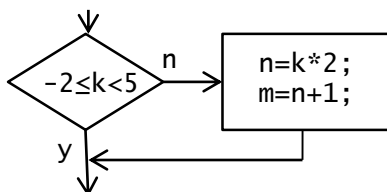
b) Assuming that **int i, k, n, m;** already declared write the following C code using if statements

```c
switch  (k) {
  case -4: case -3: case -2:
  case -1: n =m; ++m; break;
  case  0: n +=m; break;
  default: m =0;
}
```

c) Assuming that **int i, k, n, m;** already declared write C code for the following flowcharts

i)

ii)

d) Write the if-then-else structure of the following code using a switch-case statement by using the expression **(g>=95)+(g>=85)+(g>=75)+(g>=50)**.

```c
#include <stdio>
int main(void){
int n, g;
printf("Grade :");scanf("%d",&g);
if(g>=95) println("perfect");
else if(g>=85) println("excellent");
else if(g>=75) println("good");
else if(g>=50) println("passed");
else println("failed");
return 0;
}
```