Question 1 (COMPULSORY) [60 marks]

Answer all parts (a) – (t). Each part carries 3 marks.

(a) What is the screen output of the following fragment of C code (no explanation required):

```
double x=42.58;
printf("value is %7.3f\n",x);
```

value is 42.580

(b) Suppose that x, y, z and w are all of type int. If the value of w is 9, x is 20, the value of y is 3, and the value of z is 2, what is the value of w in the expression? (4)

```
w /= x / (y + y * z)
   9  20  3  3  2
```

$w = 9/(\frac{20}{9}) = 9 \times \frac{9}{20} = \frac{81}{20} = 4....$

$w = w / \left(\frac{x}{y+(y*z)}\right) \rightarrow \frac{20}{9}$ (y*z = 6)

(c) What is the screen output of the following fragment of C code (no explanation required):

```
float number = 435.73810, med ;
med = ((int)(100*number))/100.0 ;
printf("number is %7.4f\n",med);
```

$\frac{43573}{100.0} = 435.7$

435.7300

(d) What is the screen output of the following fragment of C code (no explanation required):

```
int i=0,j=1;
for (i=6;i>2;i--){
  j*=i;
  printf("j is %d\n",j);
}
```

#1 j = j * i = 1 * 6 = 6
#2 j = 6 * 5 = 30
#3 j = 30 * 4 = 120
#4 j = 120 * 3 = 360

j is 6
j is 30
j is 120
j is 360

(e) What is the screen output of the following fragment of C code (no explanation required):

```
double x=3.14159;
printf("value is %.4f",-x);
```

value is −3.1416

(f) What is the screen output of the following fragment of C code (no explanation required):

```
int i=12;
while (i>6){
  i -= 2;
  printf("i is %d\n",i);
}
```

#1 → i = 12 − 2 = 10
#2 → i = 10 − 2 = 8
#3 → i = 8 − 2 = 6
~~#4 → i = 6~~

i is 10
i is 8
i is 6

(g) What is the screen output of the following fragment of C code (no explanation required):

```
#include <stdio.h>
int f1(int a, int b, int c){
        return ((a+b)*c);
}
int f2(int a, int b){
        int c = f1(a, b, 3);
        return c+1;
}
int main(void)
{
        printf("result is %d\n", f1(-1,f2(2,4),2));
        return 0;
}
```

result is 36

(h) What is the screen output of the following fragment of C code (no explanation required)?

```
int x = -3;
int y = 2;
int* p = &y;
*p = (*p)*x + (*p)*y;
printf("x is %d and y is %d\n", x, y);
```

*p = 2 × (-3) + 2 × 2
= −6 + 4 = −2

x is −3 and y is −2

(i) What is the screen output of the following fragment of C code *(no explanation required)*?

```
char str[]="abcdefghijklmn";
char wanted[]="cdhjln";
int i, j;
for (j=0; wanted[j]!='\0'; j++){
  for (i=0; str[i]!='\0'; i++){
    if (str[i]==wanted[j]){
       str[i]='W';
       break;
    }
  }
}
printf("string=%s\n", str);
```

(m) What is the screen output of the following fragment of C code (no explanation required):

default
34
4
default
1112
default

```
int i;
for (i=1;i<15;i++){
        switch(i)
    {
    case 3:   printf("3");
        case 4: printf("4\n");
    break;
        case 11: printf("11");
        case 12: printf("12\n");
    default: printf("default\n");
        i+=i;
        }
}
```

i = 1, 3, 4, 5, 11

i = i+1 → i = 2, 10, 22

(n) What is the screen output of the following fragment of C code (no explanation required):

```
int a=2, b=0, c=-2;
if (a&&b||c){
  printf("Condition is true\n");
} else {
     printf("Condition is false\n");
   }
```

(o) What is the screen output of the following fragment of C code (no explanation required):

```
int j=0, y;
y = ! j;
printf("y is %d\n",y);
```

(p) What is the screen output of the following fragment of C code (no explanation required):

```
int nstars = 5, stars;
while (nstars >= 1) {
    stars = 1;
    while (stars <= nstars) {
      printf("*");
      stars++;
    }
    printf("\n");
    nstars--;
}
```

(q) What is the screen output of the following fragment of C code (no explanation required)?

```
int i=-5,j=-i;
if (j<=-10){
  printf("first\n");
} else if ((-i)>=(-j)){
   printf("second\n");
  } else {
     printf("no match\n");
    }
```

(r) What is the screen output of the following fragment of C code, given that the initial values in datafile.txt are:

0.2	80
0.7	60
0.1	50

avg = 16 ← #1 (0.2)(80) = 16
avg = 58 ← #2 (0.7)(60) = 42
avg = 63 ← #3 (0.1)(50) = 5

```
float prob, avg=0.0;
int quantity, num_values=0;
FILE *fptr;
fptr=fopen("datafile.txt", "r");
while(fscanf(fptr, " %f %d", &prob, &quantity)==2){
/* if return value from this fscanf() is not 2, */
/* the end-of-file indicator has been reached */
avg = avg + (prob*quantity);
}
fclose(fptr);
printf("average value is %.2f\n", avg);
```

w, a

average value is 63.00

(s) What are the contents of datafile.txt after the execution of this fragment of code, given that the initial values in datafile.txt are:

200
-54

```
int var1, var2;
FILE *fp;
FILE *fptr;
fp=fopen("datafile.txt", "r");
fscanf(fp, "%d %d", &var1, &var2);
fclose(fp);
fptr=fopen("datafile.txt", "w");
fprintf(fptr, "%d\n%d", var2, var1);
fclose(fptr);
```

(t) What is the screen output of the following fragment of C code (no explanation required)?

```
int count(int A[], int size, int target) {
  int n = 0, i;
  for (i=0; i<size; i++) {if (A[i]==target)
{n++;}}
  return n;
}

int main(void)
{
int name[]={4,6,5,3,2,5,6,7,8,9,1,3,4,5};
int result=0;
result = count(name,12,5);
printf("output is %d\n", result);

        return 0;
}
```

QUESTION 2

(a).
```
int sum = 0, i;
for(i=1 ; i<=10 ; i++)
{
    sum += i;
}
```

(b).
```
#include <stdio.h>
int zerofinder(int A[], int size) ;    // prototype
void main(void) {
    int i, array2[8] = {1,-1,-1,0,1,0,-1,1} ;
    i = zerofinder(array2, 8) ;
                .
                .
                .
}
int zerofinder (int A[], int size)
{   int j ;
    for(j=0 ; j<size ; j++)
    {   if(A[j] == 0)
        {   return j ;  }
        else
        {   return -1 ; }
    }
}
```

QUESTION 3

(a). (i). "hello there" has length 11

(ii).

```
#include <stdio.h>

int stringlength ( char s[] ) ;

void main (void)
{
    char message[80] = "hello there" ;
    printf( " \"%s \" has length %d \n", message, stringlength
                                                  (message)) ;
}

int stringlength ( char s [] )
{
    int count = 0 ;
    while (s[count] != '\0')
    {    count ++ ;    }

    return count ;
}
```

QUESTION 3

(b)

```
#include <stdio.h>

void convert(int time, int *phrs, int *pmins, int *psecs);

void main(void)
{
    int t = 0;
    int hours, mins, secs;

    printf("Please enter a time in seconds: ");
    scanf("%d", &t);

    convert(t, &hours, &mins, &secs);
    printf("The input time is %d \(%d hours, %d minutes,
        %d seconds.\).\n", t, hours, mins, secs);
}

void convert(int time, int *phrs, int *pmins, int *psecs)
{
    *phrs = time / 3600;
    time = time - (*phrs * 3600);
    *pmins = time/60;
    time = time - (*pmins * 60);
    *psecs = time;
}
```