Section 1: Single Choice(2 marks for each item, total 20 marks)

2. Given the declaration: ***int s[3][3]={1,2,3,4,5,6,7,8,9};*** the value of expression ***s[0][1]*** is equivalent to_____.
   A．s[2][0]-1　　B．s[-1][2]　　C．s[2][-1]　　D．s[1][-2]
3. Which of the following expressions is meaningful(有意义的)?_____.
   A．"hello"*2　　B．'w'*'h'　　C．"hello"[1]　　D． "hello"-'h'
4. The following code fragment will output_____.
   ```
   int n=1;
   char ch='\012';
   printf("%d", ch*n++);
   ```
   A．10　　B．12　　C．20　　D．24
5. For the declaration: ***static int a[5][ ]={0};*** Which of the following is correct?_____.
   A. The initial value of element ***a[0][0]*** is zero.
   B. The initialization is not correct in syntax.
   C. Each element in array ***a*** is initialized, but some of the values are not zero.
   D. The total number of static array ***a*** is 5.
6. If we want to open a text file ***test.txt*** under the folder ***user*** in ***C diskette***(C 盘) for the usage of both ***read*** and ***write***, which of the following statements is correct?_____.
   A．fopen("C:\user\test.txt","r");　　B. fopen("C:\user\test.txt","r+")
   C．fopen("C:\\user\\test.txt","r")　　D. fopen("C:\\user\\test.txt","r+")
7. Which function in ***string.h*** library should be used to connect two strings?_____.
   A．strlen()　　B． strcmp()　　C．strcat()　　D．strcpy()
8. Which function definition below is correct?_____.
   A. double fun(int x,int y) {z=x+y;return z;}
   B. fun(int x,y) {int z; return z;}
   C. fun(x,y) {int x,y; double z; z=x+y;return z;}
   D. double fun(int x,int y) {double z; z=x+y;return z;}
9. We want to express the meaning of "x is not equal to either 2 or 3". In the following

expressions,_____is NOT correct.

A. x != 2 || x != 3 B. !(x == 2 || x == 3)

C. x != 2 && x !=3 D. !(x == 2) && !(x==3)

10. Given: ***int *p;*** which of the following statements is ABSOLUTELY correct?_____.

A. *p = 0; B. p = 0;

C. scanf("%d", p); D. scanf("%d", &p) ;

Section 2: Fill in the blanks (2 marks for each item, total 30 marks)

1. Given: ***int a=1,b=2,c=3,d=4;***, the value of the expression ***a<b?a:c<d?c:d*** is____.
2. Given: ***char c;***, the expression_____can be used to determine that ***c*** is a digital character.
3. Given: ***int m=5,y=2;***, the value of expression ***y+=y-=m*=y*** is_____.
4. The value of expression ***!("01/24/2019"+5)[5]*** is_____.
5. The following code fragment prints out_____.

```
int i=101;
printf("%d", (i++)/2);
```

6. Given: ***char s[]="abc", *p=s;***, the value of expression ****p++*** is_____.
7. If ***x=1*** and ***y=2***, after calling ***f(&x,y)*** and ***f(&y,x)***, the values of ***x*** and ***y*** are_____.

```
void f(int *a, int b)
{
  static int k = 0;
  *a += ++k;
  b += 2;
}
```

8. Given: ***short s[][5]={301,302,303,304,305,306,307,308,309,0};*** , the values of ***sizeof(s)*** and ***strlen((char *)s)*** will be_____respectively.
9. The statement ***printf("%%d%d", 012);*** will print out_____.
10. The following code fragment will output_____.

```
void Plus(int *px) { px++;}
int x = 0; Plus(&x);
printf("%d", x);
```

11. After the following code fragment is executed, the value of ***s*** is_____.

```
int a=1, b=2, s=0;
switch (a>b) {
    default: switch(s)
        { case 0:s+=1;
        default:s+=2;break;
    }
    case 1: s+=3; break;
}
```

12. The following code fragment prints out_____.

```
int x[5]={2,4,6,8,10},*p1=&x[1], *p2=&x[4];
printf("%d" , p2-p1);
```

13. The following code fragment prints out_____.

```
int x=-1;
printf("%d",(unsigned int)x );
```

14. The following code fragment will print out_____.

```
int c[]={1, 7, 12}, *k=c;
printf("%d",*++k);
```

15. Given: ***int a=3,b=2,c=1,f;***, the value of expression ***f=a>b>c*** is_____.

Section 3: Read each of the following programs and answer questions (5 marks for each item, total 30 marks)

1. The output of the following program is_____.

```
#include <stdio.h>
#include <string.h>
int main()
{
  int a[3]={1,2,0},i,k;
  char t,s[100]="Computer Science";
  for (i=0; i<strlen(s)/3;i++){
      k=i*3;
      t=s[k];
      s[k]=s[k+a[0]];
      s[k+a[0]]= s[k+a[1]];
      s[k+a[1]]=t;
  }
  printf("%s",s);
}
```

2. The following program will output_____.

```
#include <stdio.h>
#include <string.h>
void strf1(char *dest, char *src)
{
    while(*dest) dest++;
    while(*dest++ = *src++);
}
void strf2(char *dest, char *src)
{
    int i,j,len;
    len = strlen(src);
    for(i = 0, j = 0; i < len; i += 2, j++) dest[j] = src[i];
    dest[j] = '\0';
}
int main()
{
    char a[]="Computer", s1[30],s2[30];
    strf2(s1,a);
    strf2(s2,a+1);
    strf1(s1,s2);
    printf("%s %s",s1,s2);
}
```

3. When input: ***10 -3 20 -1 40 0<ENTER>***, The following program will output_____.

```
#include <stdio.h>
#define MAX 100
#define Bottom -10
int stack[MAX];
int top;
int pop() { return stack[top--]; }
void push(int op) { if (top<MAX-1) stack[++top]=op; }
int onTop() { return stack[top]; }
int main()
{
   int n;
  top= 0;  stack[top]=Bottom;
  scanf("%d", &n);
  while (n!=0){
     if (n>0) printf("%d ",n);
```

```
        else {
              while (n<=onTop()) printf("%d ",pop());
              push(n);
        }
        scanf("%d",&n);
    }
    while (onTop()!=Bottom) printf("%d ",pop());
}
```

4. The text file ***alg3.txt*** has content as follows:
   abc***<ENTER>***
   def gh***<ENTER>***
   Then the output of the following program is_____.

```
#include <stdio.h>
int main ()
{
   FILE *fp;
   int nchars, nwords, nlines,lastnblank;
   char c;
   if((fp=fopen("alg3.txt","r"))==NULL){
      printf("Error fopen!\n"); return -1;
   }
   nchars=nwords=nlines=lastnblank=0;
   while((c=getc(fp))!=EOF) {
      nchars++;
      if(c=='\n'){
         if(lastnblank) nwords++;
         printf("%d#%d#", nwords, nchars);
         nchars=nwords=lastnblank=0;
         nlines++;
      } else {
         if(((c==' ')||(c=='\t'))&&(lastnblank)) nwords++;
         lastnblank=((c!=' ')&&(c!='\t'));
      }
   }
   printf("%d#", nlines);
   fclose(fp);
}
```

5. The following program will output_____.

```
#include <stdio.h>
void fun(int *a, int num)
{
    int *t,k;
    t = a + num - 1;
    while (a < t) { k = *a; *a = *t; *t = k; a++; t--; }
}

int main()
{
    int a[10]={1,2,3,4,5,6,7,8,9,10}, i;
    fun(a+2, sizeof(a)/sizeof(a[0])-3);
    for ( i=0; i<10; i++ ) printf("%d#",a[i]);
}
```

6. When input: ***Hello,world!#<ENTER>*** , the following program will output_____.

```
#include <stdio.h>

int IsU(char c) { return (c >= 'A' && c <= 'Z'); }
int IsL(char c) { return (c >= 'a' && c <= 'z'); }
```

```
int main(void)
{
   char c;
   while(1){
      c =  getchar();
      if(c == '#') break;
      if(IsU(c)) printf("%c", c-'A'+'a');
      else if(IsL(c)) printf("%c", c-'a'+'A');
      else printf("%c", c);
   }
}
```

Section 4: According to the specification, complete each program (2 marks for each blank, total 20 marks)

1. When enter ***n*** pairs of integer ***begin end***, the following program will output the number of natural numbers which can not be coverd and the largest one covered by the ***n*** [***begin, end***] intervals in the [***0, MAXNUM-1***] interval(输出在***[0,MAXNUM-1]***区间中未被这 n 个***[begin,end]***区间覆盖的自然数个数以及最大一个覆盖的数). For example, enter ***3 10 20 5 12 30 55*** (i.e 3 sets of intervals [10, 20], [5, 12], [30, 55]), the output is: ***count: 58, last: 55***. Fill in the blanks to complete the program.

```
#include <stdio.h>
#define MAXNUM 100

int main()
{
   int i, j, n,_____(1)_____,last=-1;
   int flag[MAXNUM];

   for (i=0; i<MAXNUM; i++) flag[i]=0;
   scanf("%d",_____(2)_____);
   for (i=0; i<n; i++) {
      int begin, end;
      scanf("%d%d",&begin, &end);
      for (j=begin;_____(3)_____; j++) flag[j]=1;
   }
   for (i=0; i<MAXNUM; i++)
      if (!flag[i])_____(4)_____;
      else last =_____(5)_____;
   printf("count:%d, last:%d", count,last);
   return 0;
}
```

2. There is a text file ***a.txt*** which contains some lines of integer array recording the performance of students (<=100 lines). And in each line, it logs **ENGLISH**, **MATH**, **SCI**, and **LIT** scores in sequence. The following program try to read in the **MATH** scores and sort them into the **Standard Output**.
   For example, suppose the file ***a.txt*** contains lines like:
   12 ***40*** *9 8****<ENTER>***
   56 ***80*** *33 77****< ENTER >***
   66 ***32*** *120 99****< ENTER >***
   66 ***20*** *120 99****< ENTER >***
   And the second column is for the math score. After execution, the following program will output as follows:
   20#32#40#80#
   Fill in the blanks to complete the program.

```
#include <stdio.h>
#define MaxSize 100

int ReadinNums(FILE *fp, int num[])
{
   int count = 0;

   while (1) {
        int math, k;
        k = fscanf(fp, "_____(6)_____", &math);
        if (_____(7)_____)  num[count++] = math; else  break;
   }
   return count;
}

void Sort(int num[], int n)
{
      int i, k, index, temp;

      for (i = 0; i < n-1; i++) {
           _____(8)_____;
           for (k = i+1; k < n; k++) {
                if (num[k] < num[index]) index = k;
           }
           if (index != i) {
               temp = num[i]; num[i] = num[index]; num[index] = temp;
           }
      }
}

void PrintNums(FILE *fp, int num[], int n)
{
      int i;

      for (i = 0; i < n; i++) fprintf(fp, "%d#", num[i]);
}

int main()
{
      int num[MaxSize], n, i; FILE *fpin, *fpout;
      if ((fpin = fopen("a.txt", "r")) == NULL) { fprintf(stderr, "Can't open file:
           a.txt\n"); return -1;
      }
      _____(9)_____;

     n = ReadinNums(fpin, num); Sort(num, n); PrintNums(fpout,
     num, n);
      _____(10)_____; /*Close file a.txt*/ return 0;
}
```

Section 1: Single Choice(2 marks for each item, total 20marks)

1_ A	2_ D	3 C	4 A	5_ B
6_ D	7_ C	8 D	9 A	10_ B

Section 2: Fill in the blanks (2 marks for each item, total 30marks)

1	1	2	c > =‘0’ && c <= ‘9’
3	-16	4	1
5	50	6	‘a’
7	2, 4	8	20, 18
9	%d10	10	0
11	6	12	3
13	-1	14	7
15	0		

Section 3: Read each of the following programs and answer questions (5 marks for each item, total 30 marks)

1 omCutpr eciSncee

2 Cmueoptr optr

3 10 20 40 -1 -3

4 1#4#2#7#2#

5 1#2#9#8#7#6#5#4#3#10#

6 hELLO,WORLD!

Section 4: According to the specification, complete each program (2 marks for each blank, total 20 marks)

(1) count=0 (2) &n

(3) j<=end (4) count++

(5) i (6) %*d%d%*d%*d

(7) k==1 (8) index=i

(9) fpout=stdout (10) fclose(fpin)