子程式	呼叫方法

1. int min (int x, int y){
    if(_____
        return _____
    else return _____
}
	scanf ("%i%i", &a, &b);
	n = min(a,b);
	if (c < min(a,b))
	    printf ("c is min \n");

2. int leapYear (int yy){
    if (_____
 	
        return _____
    else
        return _____
}
	scanf ("%i", &yr);
	n = leapYear(yr);
	if(n==1 && mm==2)
	    days=29;

3. int sumN (int n){
    int i, total= _____
    for (i=1;_____
        total = _____
    return _____
}
	// n = 1+2+3+…+10
	n = sumN (10);

4. void drawline (char c, int n){
    int i;
    for (i=0;_____
        printf (_____
    printf ("\n");
}
	drawline('*',40);
	****** … ******
	drawline('–',50);
	drawline('~',60);

5. char upcase (char c){
    if (_____
        return _____;
    else
        return _____;
}
	char house='m'; // "M"
	house = upcase (house);
	house = toupper (house);

6. int maxdays (int mm){
    switch(_____){
        case 2: return 28; break;
        case _____
        case _____
            return _____
        default: return _____
    }
}
	n = maxdays(1); // 31
	n = maxdays(2); // 28
	n = maxdays(4); // 30

子程式	呼叫方法

7. float ctof (float c){
　　return ＿＿＿＿
}
	// °C 轉°F　　$f = \dfrac{9}{5}c + 32$
	
	f = ctof(100);

8. float ftoc (float f){
　　return ＿＿＿＿
}
	// °F 轉°C　　$c = \dfrac{5}{9}(f - 32)$
	
	c = ftoc(212);

9. char score2grade (int n){
　　if (n<30) return 'F';
　　else if ＿＿＿＿
　　else if ＿＿＿＿
　　else if ＿＿＿＿
　　else if ＿＿＿＿
　　else return 'A';
}
	// 分數轉等級
	int score=88;
	char grade='A';
	grade = score2grade (score);
	
	// F(<30),E(<50),D(<60)
	// C(<70),B(<80),A(<=100)

10. int ndigits (int x){
　　int n=0;
　　x = abs(x);
　　while (＿＿＿＿＿＿＿){
 	
　　　＿＿＿＿
　　　＿＿＿＿
　　}
　　return n;
}
	// 多少個數位
	n = ndigits(1234);　　// 4
	n = ndigits(13579);　// 5

11. int digitSum (int x){
　　int sum=0;
　　while (＿＿＿＿＿＿＿){
 	
　　　＿＿＿＿
　　　＿＿＿＿
　　}
　　return ＿＿＿＿;
}
	// 數位之和
	n = digitSum(1234);　// 10
	n = digitSum(13579);　// 25

12. int nthdigit (int x, int n){
　　char s[10];
　　itoa (x,s,10);
　　if (　＿＿＿＿
　　　　return -1;
　　return (　＿＿＿＿
}
	// 第 n 個數位
	n = nthdigit (13579,2); // 3
	n = nthdigit (13579,3); // 5
	n = nthdigit (13579,9); // -1

子程式	呼叫方法

13. int confirm (char msg[]){
    char c='N';
    do{
        printf (_____
        scanf (_____
        c = toupper(_____
 	
        if (_____
            printf ("Error\n");
    }while(_____
 	
    if (_____
    else return 0;
  }
	n = confirm ("Student");
	n = confirm ("Hungry");
	n = confirm ("Continue");
	
	Student <y/n>? y
	Hungry <y/n>? n
	Continue <y/n>? y

14. int getNum (char msg[], int n1, int n2){
    int num;
    do{
        printf (_____
        scanf (_____
        if (_____
            printf ("Error\n");
    }while(_____
 	
    return _____
  }
	dow = getNum ("星期",0,6);
	mm = getNum ("月份",1,12);
	mark = getNum ("分數",0,100);
	age = getNum ("年齡",1,90);
	hour = getNum ("時間",0,23);

15. char getChoice (char frCh, char toCh){
    char c='x';
    do{
        printf (_____
        scanf (_____
        c = toupper(   );
        if (_____
            printf ("Error\n");
    }while(_____
 	
    return _____
  }
	char choice;
	choice = getChoice ('X','Z');
	choice = getChoice ('0','5');
	choice = getChoice ('A','F');
	
	ch = getChoiceFrom ("AEIOU");
	??

16. int digitOf (int x, int n){
   if (x<n) return -1;
   return _____
}
	// 某個數位 (個十百千...位)
	n = digitOf (13579,100); // 5
	n = digitOf (135,10000); // -1

17. int discriminant (int a, int b, int c){
   return _____
}
	// $ax^2+bx+c=0$ $d=b^2-4ac$
	n = discriminant(1,4,2); // 8

18. int noOfRoots (int a, int b, int c){
   int d = discriminant(a,b,c);
   if(d<0) return _____
   if(d==0) return _____
   if(d>0) return _____
}
	// $Ax^2+Bx+C=0$ x=??
	// 有多少個根 roots
	n = noOfRoots(1,4,2); // 2
	n = noOfRoots(1,2,1); // 1
	n = noOfRoots(1,0,1); // 0

19. void displayMenu(){
   printf ("增新 Add   \n");
   printf ("刪除 Delete   \n");
   printf ("找尋 Search   \n");
   printf ("修改 Modify   \n");
}
	// 顯示選單
	displayMenu();
	
	// "離開 Quit

20. void swap (int *a, int *b){
   int t;
   t = *a;   // a (pointer/address)
   *a = *b;   // *a (content)
   *b = t;
}
	int x=3, y=4;
	// 數值互換
	swap (&x, &y);
	
	&x (address of x)

21. void rotateLeft (int array[], int max){
   int i;
   for(i=0;_____
      array[i] = _____
}
	int num[10]={0,1,2,...};
	// 左移
	rotateLeft(num,10);

22. void rotateRigft (int array[], int max){
   int i;
   for(i=_____
      array[i] = _____
}
	// 右移
	rotateRight(num,10);

23. void revArray (int array[], int max){
   int i;
   for(i=0;_____
      swap(_____
}
	// 倒轉陣列 num[]
	reverseArray(num,10);

24. float hypotenuse (int a, int b){
　　return ＿＿＿＿
}
	// 直角三角形斜邊
	c = hypotenuse(3,4);

25. int isTriangle (int a, int b, int c){
　　if ＿＿＿＿
　　　　return 1;
　　return 0;
}
	// 是否三角形
	n = isTriangle (3,4,5);　// 1
	n = isTriangle (3,4,7);　// 0

26. int reverseNum (int n){
　　int x=0;
　　while (n>0){
　　　　x = ＿＿＿＿
　　　　n ＿＿＿＿
　　}
　　return x;
}
	// 倒序
	// 12345 -> 54321
	n = reverseNum (12345);

27. int random (int n1, int n2){
　　return rand()% ＿＿＿＿
}
	// 隨機數
	n = random(10,20);

28. int isprime (int n){
　　int i;
　　for (i=2; i<=＿＿＿＿
　　　　if ＿＿＿＿
　　return 1;
}
	// 是否質數
	n = isprime(13);　　// 1
	n = isprime(100);　// 0

29. int parity (char s[], int par){
　　int i, sum=0;
　　for (i=0; i<＿＿＿＿
　　　　if ＿＿＿＿
　　return ＿＿＿＿
}
	// 奇偶校驗位
	n = parity("1001",1);　// 1
	n = parity("1001",0);　// 0

30. void moneyChange (int amt){
　　int i=0;
　　while(amt>0){
　　　　if (amt ＿＿＿＿
　　　　　　＿＿＿＿
　　　　　　＿＿＿＿
　　　　}else＿＿＿＿
　　}
}
	int note[9]= {1000,500,100,
	　　　　　　50,20,10,5,2,1};
	int freq[9]={0};
	int amount=1234;
	
	moneyChange (amount);

31. `void num2roman (int n, char s[]){`
    `int i=0;`
    `s[0] = '\0';`
    `while (n>0){`
        `if (n _____`
          `_____`
          `_____`
        `}else_____`
    `}`
`}`
	`int num[13] = {1000,900,500,`
	`400,100,90,50,40,10,9,5,4,1};`
	`char roman[MAX][3]`
	    `={"M","DM","D","CD",`
	        `"C","XC","L","XL","X",`
	        `"IX","V","IV","I"};`
	`// 轉羅馬數字`
	`num2roman (23, romanStr);`
	`puts(romanStr);`

32. no.of days From Jan 1 to dd-mm-yy
`int ndays (int mm, int dd){`
    `int i, n= _____`
    `for (i=1; i<_____`
        `n _____`
    `return n;`
`}`
	`int maxdays[13] = {`
	    `0,31,28,31,30,31,30,`
	    `31,31,30,31,30,31};`
	`n = ndays (12,3);`
	    `// 31+28+...+3 = 337(天)`
	`n = ndays (3,10);`
	    `// 31+28+10 = 69(天)`

33. `int dateDiff (int yy1, int mm1, int dd1,`
            `int yy2, int mm2, int dd2){`
    `int i, n1,n2,n3=0;`
    `n1 = ndays ( _____`
    `n2 = ndays ( _____`
    `n3 = 365* _____`
    `return n3+n2-n1;`
`}`
	`// 相差日數 ,請利用 ndays()`
	`n = dateDiff`
	    `(2000,7,1,  2009,4,15);`

34. `int checkStr (char s[]){`
    `int n = _____`
    `if( _____`
        `return 1;`
    `else return 0;`
`}`
	`valid = checkStr("1x");`
	`valid = checkStr("2or");`
	`valid = checkStr("3axe");`
	`...`
	`valid = checkStr("5apple");`

35. `void fibonacci (int n){`
    `int i, a=0,b=1,c=1;`
    `for (i=0;_____`
        `c = _____`
        `printf ("%i\t", c);`
        `a =  _____`
        `b =  _____`
    `}`
`}`
	`fibonacci (7);`
	`// 1,1,2,3,5,8,13,...`
	
	`fibonacci (9);`
	`// 1,1,2,3,5,8,13,21,34...`

36. void dateBritish(char s[]){
    char t[11];
    strcpy(t,s);
    s[0]=t[8];
 	
}
	char s[10]="2009-08-15";
	dateBritish(s); puts(s);
	// 15-08-2009 (dd-mm-yy)

37. void dateAmerican(char s[]){
    char t[11];
    strcpy(t,s);
    s[0]=t[8];
 	
}
	char s[10]="2009-08-15";
	dateAmerican(s); puts(s);
	// 08-15-2009 (mm-dd-yy)

38. int gcd (int a, int b){
    int r;
    while((r=_____)>0){
        a = _____
        b = _____
    }
    return b;
}
	// GCD = HCF 最大公因數
	n = gcd(12,20); // 4
	
	// LCM*HCF = a*b
	m = lcm(12,20) // 60

39. int mystrcmp (char s1[], char s2[]){
    int n,x;
    n = _____;
    for(i=0;i<n;i++){
        if(s1[i]<s2[i]) return _____
        if(s1[i]>s2[i]) return _____
    }
    n = strlen(s1); x = strlen(s2);
    if(n==x) return _____
    if(n<x) return _____
    if(n>x) return _____
}
	n = mystrcmp ("edb", "edba");
	// -1
	
	n = mystrcmp ("edb", "EDB");
	// 1
	
	n = mystrcmp ("edb", "edb");
	// 0

40. int checkDate(dd,mm,yy){
    if(mm>=1 &&_____
 	
        return _____
    return _____
}
	int dd=31, mm=11, yy=2009;
	n = checkDate (dd,mm,yy);
	n ← 0
	n = checkDate (31,12,yy);
	n ← 1

41. int checkIDno (char id[]){
    int i;
    if (strlen(id) _____
    for (i=1;i<7;i++)
        if (id[i] _____
    if (id[7]=='A' ||
 	
        _____
        return   _____
    return   _____
}
	char idno[9]= "A1234567";
	n = checkIDno (idno);
	n ← 0
	
	n = checkIDno ("A123");
	n ← 0
	
	n = checkIDno ("A1234563");
	n ← 1

42. char checkDigit (char id[]){
    int i,j=7,sum=58*9 + _____
    for (i=1;i<7;i++_____
        sum +=   _____
    if (id[7]=='A') sum +=   _____
    else sum +=   _____
 	
    j = sum%11;
    if (j==0) return   _____
    else if (j==1) return   _____
    else return   _____
}
	char idno[9]= "A1234567";
	
	if (checkIDno (idno)){
	    c = checkDigit(idno);
	
	    if(c==idno[7])
	        puts("Valid");
	    else
	        puts("Invalid");
	}

44. int symmetrical (char s[]){
    int i=0,j= _____
    while(i<j){
        if (s[i]!=s[j]) _____
        _____
    }
    return _____
}
	char s[10]="madam";
	// 是否對稱
	n=palindrome(s); // →1
	n=symmetrical(s);

45. void uppercase (char s[]){
    int i;
    for (i=0;i<_____
        s[i] = toupper _____
}
	char s[10]="madam";
	uppercase(s); // "MADAM"

46. void rtrim (char s[]){
    int n= _____
    while (n>=0 && _____
 	
    _____
}
	char s[10]="madam ";
	rtrim(s); // 刪右空白

47. char yob2grade (int yob){
　　int yy, mm, dd, age;
　　systemDate (_____
　　age = _____
　　if (age>=17) return _____
　　else _____
　　else _____
}
	Grade　age
	A　17 歲或以上
	B　15-16 歲
	C　14 歲或以下
	grade = yob2grade(1995);

48. char getChar (char msg[], char opt[]){
　　char ans; int ok=0, i;
　　do{
　　　　printf (_____
　　　　scanf (_____
　　　　if (_____
　　　　　　printf ("Error\n");
　　}while(_____
　　return ans;
}
	// 輸入字符 char
	gender=getChar ("性別", "MF");
	
	size= getChar ("Size", "LMS");
	
	grade= getChar ("等級", "ABCDEFU");
	
	dir= getChar ("方向", "ESWN");

49. void rotateLeft (char s[]){
　　int i, n=strlen(s);
　　char c= _____
　　for(i=0; _____
　　　　s[i] = _____
　　_____
}
	char name[10]="ablmcc";
	// 左移
	rotateLeft(name);

50. void rotateRigft (char s[]){
　　int i, n=strlen(s);
　　char c= _____
　　for(i=_____
　　　　s[i] = _____
　　_____
}
	// 右移
	rotateRight(name);

51. void reverseString (char s[]){
　　int i, n=strlen(s); char c;
　　for(i=0;_____
　　　　c = _____
　　　　s[i] = _____
　　　　_____
　　}
}
	char name[10]="abcde";
	reverseString (name);
	// "abcde" --> "edcba"

52. void ltrim (char s[]){
　　int i=0, n= _____
　　while (i<n && _____
 	
　　　　_____
}
	char s[10]="　　　madam";
	ltrim(s);　　// 刪左空白

53. void alltrim (char s[]){
　　_____
 	
}
	利用 ltrim() 及 rtrim()
	刪去左右空白

54. int wordCount(char s[]){
　　int i, inword=0, count=0;
　　for (i=0; i<strlen(s); i++){
　　　　if (_____
　　　　　　if (_____
 	
　　　　　　　　_____
 	
　　　　　　　　_____
　　　　　　}
　　　　} else
 	
　　　　　　_____
 	
　　}
　　return count;
}
	n = wordCount("ab lm cc");
	// 數字數 n=3

55. void replace (char s[], char t, char x){
　　int i=0, n=strlen(s);
　　for (i=0; _____
　　　　if (_____
}
	// 'a'取代為'A'
	replace("banana",'a','A');
	// "bAnAnA"

56. int maxdays (int yy, int mm){
　　switch(mm){
　　　　case 2: return 28+ _____
　　　　　　　break;
　　　　case 4: _____
　　　　case 9: _____
 	
　　　　　　_____
 	
　　　　　　_____
　　}
}
	// 月尾
	n = maxdays(2000,12); // 31
	n = maxdays(2010,2); // 28
	n = maxdays(2008,2); // 29

57. float median (int arr[], int n){
　　if(n%2==1) return arr[___];
　　return (arr[___]+arr[___])/2.0;
}
	int age[]={12,12,13,14,15};
	n = median(age,5);　　// 13
	中位數

58. void shuffleCards(){   // 洗牌 1
　　for(n=0;n<52;n++){
　　　　x = rand ____
　　　　swap(_____
　　}
}
	int card[52]={0,1,2,3,...};
	for(n=0;n<52;n++)
	　　card[n]=n;
	
	洗牌後: card[]={15,3,40,29,...}

59. void shuffleCards2(){   // 洗牌 2
　　for(n=0;n<500;n++){
　　　　x = rand____
　　　　y = rand____
　　　　swap(_____
　　}
}
	同上

60. char cardName (int cardno){
　　char s[]="1234567890JQK";
　　int n = ____
　　return s[n];
}
	char cn='A';
	cn = cardName(0); // 'A'
	cn = cardName(9); // '0'
	cn = cardName(10); // 'J'
	cn = cardName(11); // 'Q'
	cn = cardName(51); // 'K'

61. char suitName (int cardno){
　　char s[]="CDHS";
　　int n = ____
　　return s[n];
}
	clubs♣, diamonds♦, hearts♥, spades♠
	
	sn = suitName(card[0]);

62. int cardPoints (int cardno){
 	
}
	11-13(10)

63. polynomial generator: -2(3x+4)	

64. int constant(char s[]){
　　int i, n=strlen(s);
　　i=n-1;
　　while(s[i]
}
	n = constant("23x+14y+56");

65. n = coefficientX("23x+14y+56"); 　　n = coefficientY("23x+14y+56"); 　　void insertItemAt (int item, int n)	int stackFull() int stackEmpty() void stackPush(int item) int stackPop()
66. int queueSize() 　　int queueFull() 　　int queueEmpty()	void enqueue(int item) int dequeue()

int listFull() int listEmpty() void traverse()	

本利和
amt = cmpdInterest(p,r,n);
amt = simpleInterest(p,r,n);
pay = monthlyRepayment(p,r,n);
Given: p,r,pay;   find n
n = doubleDeposit(p,r)
no.of months to make 2p	

evaluate("4+5-3") spaces(10) formatNumber() num2words	

n = mystrlen(s);
n = mystrcmp(s1,s2);
mystrcpy(s1,s2);
mystrncpy(s1,s2,n);	

int interval[]={12,31,19}; float rate[]={0,4.16,6.45,9.05}; pay = waterBill(50);	
int validPassword() float average (int array[]) void barchart (int array[]) void histogram (int array[])	
void bubblesort(int array[]) int linearsearch(int array[], int target) int binarysearch(int array[], int target) void bubblesort(char array[][]) int linearsearch(char array[][], char target[]) int binarysearch(char array[][], char target[]) excel functions void left (char source[], int n, char target[]) void right (char source[], int n, char	

target[]) void substr(char source[], int p, int n, char target[]) int findpos(char source[], char target[])	

char result[10];
 left   ("abcdef",3,   str); // str = "abc"
 right ("abcdef",3,   str);   // str = "def"
 substr("abcdef",3,2,str);   // str = "de"
 n=substrpos("abcdef","cd");   // n = 2
 	
 void drawTriangle(int n)
 *
 * *
 * * *
 * * * *
 * * * * *
 int typhoon(int dist)
 signal = typhoon(distance from HK)	

void printCalendar(int yy, int mm)
 int myrand()
 int tossCoin()
 int tossCoins(int n) // toss n coins
 int roll2dice()
 	
 int countOccurrence(int array[], int n)
 float repayment(float p, int n, float r)
 	
 int myabs(int n)
 int isodd(int n)
 int iseven(int n)
 int nCr(int n, int r) // = n!/(n-r)!r!	

void hex2bin (char hexStr[], char binStr[])
    // "1A" -> "1 1010"
 void bin2hex (char binStr[], char hexStr[])
    // "1 1010" -> "1A"
 void dec2bin (int dec, char binStr[])
    // 26 -> "1 1010"
 void dec2hex (int dec, char hexStr[])
    // 26 -> "1A"	

char lastChar (char s[]) // 傳回字串 s[]的最後一個字符	ch = lastChar ("abcde"); // ch = 'e';
char randLetter () // 隨機地傳回一個英文字母 (A-Z)	ch = randLetter();
char randLetter (char s[]) // 隨機地傳回一個英文字母(按指定範圍)	ch = randLetter("aeiou");
n = zodiac(mm,dd);	
加油 加加油 加加加油	printText("加油");

p = priceOfFood(food);
char c = randLetter();
char c = randLetters("aeiou");
c = myupper(c);
rewrite n = minimum(a,b,c,d,e,f) using min	

c = last("aeiou");
n = findpos(s, 'x');
n = whowins ();   // paper-scissors-stone
n = endGame (); // tic-tac-toe	

unix2dos(char s[])
mac2dos
dos2unix
list_dir(path)
get_curr_dir() getcwd()
gif2html
file_compare(f1,f2)
alpha_numeric(s)   // eliminate all non-alpha-numeric chars from s
                deletes all the white spaces, punctuations and other special
                characters leaving and displaying only the letters and digits.
void factorsOf (n)   // factorsOf(12)：1,2,3,4,6,12
for(i=2;i<?;i++) if(n%i==0) printf("%i",i);

// no.of working days(excluding Sundays & Saturdays)
n = workdays(yy1,mm1,dd1,yy2,mm2,dd2);

```int isAsc (char s[]){ int i=1, ok=1; while(ok && i<strlen(s)){ if(s[i-1]>s[i]) ok=0; i++; } return ok;}```	`// in Ascending order?`
	`n = isAsc("abcde");   // 1`
	`n = isAsc("acexy");   // 1`
	`n = isAsc("edcba");   // 0`
	`n = isAsc("abcba");   // 0`

`int isDesc (char s[])`	
```int inOrder (char s[]){ return (isAsc(s) || isDesc(s));}```	