計算機程式 113 學年度第 1 學期小考 1 試題

系級：_____學號：_____姓名：_____

(Each 3%)

1. Please list 3 advantages about applying function module.

利於團隊分工，縮短程式開發時間

縮短程式長度，增加可讀性，方便測試驗證、除錯、重複使用

開發類似功能產品時，只需稍微修改即可套用

- 使用函式的好處
  - 團隊分工：將大程式切割由多人撰寫，縮短程式開發時間。
  - 重複運用：函式可重複呼叫，縮短程式長度。
  - 提高可靠度、可讀性，利於測試驗證、除錯。
  - 後續修改使用：再開發類似功能產品，只需稍微修改即可套用。

2. __(b)(d)__Choose the legal naming variable? (multiple)
(a) else (b) _apple (c) global (d) name (e) 3NTUT

3. Output of the code: (1) 4 (2) 72

```
01  def test():
02      a = 10
03      Bb = a + 5
04      c2 =  Bb - 7
05      c2 *= 3
06      a = c2 % 10
07      print(a)      #(1)
08      print(c2*3)  #(2)
```

4. Output of test(): (1) Ann is Tom!
(2) Tom is Ann! (3) Ann is Gary!

```
01  def format00():
02      return '{1} is {0}!'.format('Tom', 'Ann')
03
04  def format01():
05      return '{A} is {B}!'.format(A='Tom', B='Ann')
06
07  def format02():
08      result = '{A} is {B}!'.format(B='Tom', A='Ann')
09      result = result.replace('Tom', 'Gary')
10      return result
11
12  def test():
13      print(format00())  #(1)
14      print(format01())  #(2)
15      print(format02())  #(3)
```

5. **Fill in the space to complete the code.** Players A and B each get five cards. They are comparing whose points are bigger. If one player gets over 21 points, the point will be reset to 0. If player A wins, print “a win”, and if B wins then print “b win”; otherwise print “tie”. The points match as follows:

symbol	A	2~10	J, Q, K
points	1	2~10	0.5

A, B 兩位玩家各得 5 張撲克牌，雙方比較總點數大小。若總點數 > 21，則總點數重設為 0 點。若 A 獲勝，輸出 a win，若 B 獲勝，輸出 b win，若平手，輸出 tie。)

(1) int(card) (2) transferPoint (3) compare

```
01  def transferPoint(card):
02      if(card == 'A'):
03          return 1
04      elif(card == 'J' or card == 'Q' or card == 'K'):
05          return 0.5
06      else:
07          return ___(1)___
08
09  def getSum(a,b,c,d,e):
10      a_pt = ___(2)___(a)
11      b_pt = ___(2)___(b)
12      c_pt = ___(2)___(c)
13      d_pt = ___(2)___(d)
14      e_pt = ___(2)___(e)
15      sum = a_pt + b_pt + c_pt + d_pt + e_pt
16      if sum > 21:
17          return 0
18      else:
19          return sum
20
21  def compare(a,b):
22      if a>b:
23          print('a win')
24      elif a<b:
25          print('b win')
26      else:
27          print('tie')
28
29  def main():
30      A1,A2,A3,A4,A5 = '2','3','5','10','A'
31      B1,B2,B3,B4,B5 = 'A','7','10','2','Q'
32      a_pt = getSum(A1,A2,A3,A4,A5)
33      b_pt = getSum(B1,B2,B3,B4,B5)
34      ___(3)___(a_pt, b_pt)
```

6.Output of test(): (1) Anny key (2) 85 155

```
01  def test():
02      names = ["Tom", "John", "Mary"]
03      scores = [85, 90, 70]
04      names = sorted(names)
05      names[0] = "Anny"
06      scores[1] = scores[0] + scores[2]
07      students = [names, scores]
08      students[0][1] = "key"
09      print(students[0][0], students[0][1])  # (1)
10      print(students[1][0], students[1][1])  # (2)
```

7. Output of (1) F(10, 8, 2, 6) E (2) F(3, 5, 4, 7) I

```
01  def F(p, q, r, s):
02      if p > q:
03          if r > s:
04              if (p + s) % 2 == 0:
05                  print("A")
06              elif (r - q) % 3 == 0:
07                  print("B")
08              else:
09                  print("C")
10          elif s == q:
11              print("D")
12          else:
13              print("E")
14      elif p == q:
15          if r * s < p:
16              print("F")
17          else:
18              print("G")
19      else:
20          if q > r:
21              if s % 2 == 0:
22                  print("H")
23              else:
24                  print("I")
```

```
25          else:
26              print("J")
```

8. (1) 3 (2) side3 (3) 1

Output of f(7, 8, 9) is 11, 18. Complete the empty spaces.

```
01  def get_perimeter(side1, side2 = (1) , (2) = (3) ):
02      return side1+ side2 + side3
03  def f(a, b, c):
04      print(get_perimeter(a))
05      print(get_perimeter(b, c))
```

9.Output of test(1, 3, 8):(1) a (2) 4 (3) 9

```
01  def test(A, B, C):
02      data = [1, 2, 3, 4, 5, 6, 7, 8, 9, 10]
03      a = 10
04      b = a * 20
05      c = b // 3
06      d = c % 2
07      if a > 10:
08          data[8] = 5
09      elif b > c:
10          data[1] = "a"
11      elif c > d:
12          data[3] = "b"
13      else:
14          data[5] = "c"
15      print(data[A])  # (1)
16      print(data[B])  # (2)
17      print(data[C])  # (3)
```

10.Output: (1) [101, 202] (2) [22, 99]

```
01  def match(sec1: int, sec2: int, secs: list, index: int)-
02  >int:
03      if sec1==sec2:
04          secs[index] = sec1
05          index = index + 1
06      return index
07
08  def isConflict(cou1:list, cou2: list):
09      result = [0, 0]
10      secs = [99, 99]
11      index = 0
12      index = match(cou1[1], cou2[1], secs, index)
13      index = match(cou1[2], cou2[1], secs, index)
14      if index>0:
15          sections = sorted(secs)
16          c_ids = [cou1[0], cou2[0]]
17          c_ids = sorted(c_ids)
18          result = [c_ids, sections]
19      print(result[0])
20      print(result[1])
21
22  courses = [[202, 11, 22],[101, 22, 55]]
23  isConflict(courses[0], courses[1])
```

11. **Fill in the space to complete the code.**

(1) input() (2) ** (3) (name,BMI)

BMI formula: BMI = weight(kg)/height^2(meter^2).

E.g. 52 kg and 155cm, BMI = 52/(1.55^2) =**21.60.**

Range	Output
BMI<18.5	too LOW
18.5<=BMI<=24	
24< BMI	too HIGH

The code must input name, height and weight from keyboard.

If BMI is too large, print "Hi {name}, BMI:{bmi, two decimal} too HIGH", if BMI is too low, print out "Hi {name}, BMI:{bmi, two decimal} too LOW")

```
01  def health ():
02      name = ___(1)___
03      weight = float(___(1)___)
04      height = float(___(1)___)
05      BMI = weight / ((height) ___(2)___2)
06      if BMI < 18.5:
07          print ("Hi %s, BMI : %.2f too LOW" % ___(3)___)
08      elif BMI > 24:
09          print ("Hi %s, BMI : %.2f too HIGH" % ___(3)___)
```

12.Output of (1) f(1): Joe (2) f(3) Sam

```
01  def f(N):
02      mylist = ["Sam", "Tom", "Cat", "Joe", "Kim"]
03      mylist = sorted(mylist)
04      print(mylist[N])
```

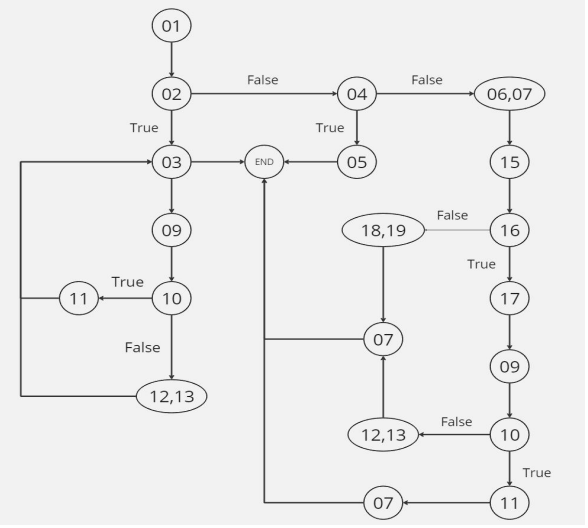
13. (1) Output of (1) f(5): 6 (2) f(10) 10

```
01  def f(arg_a):
02      if arg_a > 10:
03          return b(arg_a * 5)
04      elif arg_a > 5:
05          print(arg_a)
06      else:
07          return c(arg_a - 3)
08
09  def b(arg_b):
10      if arg_b % 2 != 0:
11          print(arg_b - 10)
12      else:
13          print(arg_b)
14
15  def c(arg_c):
16      if arg_c >= 2:
17          b(arg_c * 3)
18      else:
19          print(arg_c)
```

(2) The execution sequence of f(5):

[1, 2, 4, 6, 7, 15, 16, 17, 9 ,10 ,12, 13, 7]

(3) Draw the flow chart of the f().



14. What is the MVC: (1) M: Model

(2) V: View (3) C: Controller

15. Please describe how to learn programming well for at least 30 words. (請描述如何學好程式設計)