



Python 四级

2024 年 06 月

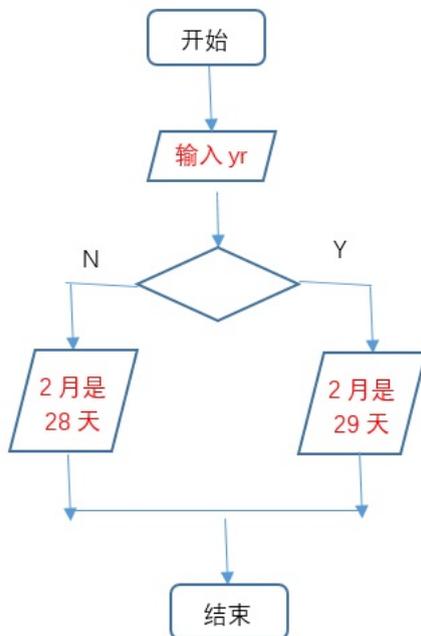
1 单选题（每题 2 分，共 30 分）

题号	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
答案	C	B	C	D	C	A	A	A	D	A	D	D	B	B	D

第 1 题 小杨父母带他到某培训机构给他报名参加CCF组织的GESP认证考试的第1级，那他可以选择的认证语言有几种？（ ）

- A. 1
- B. 2
- C. 3
- D. 4

第 2 题 下面流程图在yr输入2024时，可以判定yr代表闰年，并输出 2月是29天，则图中菱形框中应该填入（ ）。



- A. $(yr\%400==0) \ || \ (yr\%4==0)$
- B. $(yr\%400==0) \ || \ (yr\%4==0 \ \&\& \ yr\%100!=0)$
- C. $(yr\%400==0) \ \&\& \ (yr\%4==0)$
- D. $(yr\%400==0) \ \&\& \ (yr\%4==0 \ \&\& \ yr\%100!=0)$

第 3 题 执行下面Python代码后，输出的结果可能是？（ ）

```
1 name = ('Tom', 'Join', 'Gaia')
2 age = (10, 13, 12)
3 print({(a.upper(), b) for a, b in zip(name, age)})
```

- A. {'Tom': 10, 'Join': 13, 'Gaia': 12}
- B. {'TOM': 10, 'JOIN': 13, 'GAIA': 12}
- C. {('JOIN', 13), ('TOM', 10), ('GAIA', 12)}
- D. {('Join', 13), ('Tom', 10), ('Gaia', 12)}

第4题 函数fun的定义如下，调用该函数的语句错误的是？（）

```
1 def fun(x, y, z=3):
2     print(x, y, z)
```

- A. fun(1, 2, 3)
- B. fun(1, 2)
- C. fun(y=2, x=1)
- D. fun(1, y=2, 3)

第5题 执行下面Python代码后，输出的结果是？（）

```
1 def fun(x, y, *nums):
2     res = x + y
3     for i in nums:
4         res += i
5     return res
6
7
8 print(fun(10, 20, 15, 25, 30))
```

- A. 30
- B. 45
- C. 100
- D. 语法错误

第6题 执行下面Python代码后，输出的结果是？（）

```
1 def fun(x, y, **kwargs):
2     res = x + y
3     for i in kwargs.values():
4         res += i
5     return res
6
7
8 print(fun(10, 20, A=6, B=1, C=3))
```

- A. 40

- B. 36
- C. 30
- D. 语法错误

第7题 执行下面python代码后，输出的结果是？（）

```
1 def add():
2     count = 1
3
4     def fun():
5         nonlocal count
6         print(count, end="#")
7         count += 2
8
9     return fun
10
11
12 a = add()
13 a()
14 a()
```

- A. 1#3#
- B. 1#1#
- C. 3#3#
- D. add函数语法错误

第8题 执行下面python代码后，输出的结果是？（）

```
1 s = 1
2
3
4 def sums(n):
5     global s
6     s = 0
7     s = s + n
8     print(s, end="#")
9
10
11 sums(5)
12 print(s, end="#")
```

- A. 5#5#
- B. 5#1#
- C. 1#1#
- D. 1#5#

第9题 执行下面python代码后，输出的结果是？（）

```

1 def add(c):
2     d = c.append(40)
3     return d
4
5
6 a = [10, 20, 30]
7 b = add(a)
8 print(a, b)

```

- A. [10, 20, 30] [10, 20, 30]
- B. [10, 20, 30, 40] [10, 20, 30, 40]
- C. [10, 20, 30] [10, 20, 30, 40]
- D. [10, 20, 30, 40] None

第 10 题 在一个农场上，有一头母牛。根据农场的规则，这头母牛每年年初都会生出一头小母牛。每头新生的小母牛从它们出生的第四个年头开始，也会每年年初生出一头新的小母牛。假设没有母牛死亡，并且每头母牛都严格遵循这个规则。下列程序用来计算在第 n ($n < 30$)年时，农场上共有多少头母牛，其中横线处填写的代码为？（）

```

1 lst = [0] * 100
2 lst[1:4] = [1, 2, 3, 4]
3
4
5 def calculate_cows(n):
6     if n <= 4:
7         return lst[n]
8     i = 5
9     while i <= n:
10        lst[i] = _____
11        i += 1
12    return lst[i - 1]

```

- A. $lst[i - 1] + lst[i - 3]$
- B. $lst[i - 2] + lst[i - 3]$
- C. $lst[i - 1] + lst[i - 2]$
- D. $lst[-1] + lst[-3]$

第 11 题 程序段如下：

```

1 def fun(arr: list):
2     n = len(arr)
3     for i in range(n - 1):
4         for j in range(0, n - i - 1):
5             if arr[j] > arr[j + 1]:
6                 arr[j], arr[j + 1] = arr[j + 1], arr[j]

```

其中， n 为正整数，则该段程序的最坏时间复杂度是？（）

- A. $O(n)$
- B. $O(2n)$

C. $O(n^3)$

D. $O(n^2)$

第 12 题 执行下面Python代码后，输出的结果是？（ ）

```
1 users = [  
2     {"name": "Tom", "skills": {"Python", "Java", "C++"}},  
3     {"name": "Join", "skills": {"Python", "Java"}},  
4     {"name": "Gaia", "skills": {"Java", "C++", "SQL"}}  
5 ]  
6  
7 skills_sets = (user['skills'] for user in users)  
8 common_skills = set.intersection(*skills_sets)  
9 print(common_skills)
```

A. {'Java', 'Python'}

B. {'SQL', 'C++'}

C. {'Java', 'SQL', 'Python', 'C++'}

D. {'Java'}

第 13 题 执行下面Python代码后，输出的结果是？（ ）

```
1 lst = [1, 4, 3, 5, 2]  
2 lst = [i[0] for i in sorted(enumerate(lst), key=lambda x: x[1])]  
3 print(lst)
```

A. [3, 1, 2, 4, 0]

B. [0, 4, 2, 1, 3]

C. [3, 0, 2, 1, 4]

D. [0, 3, 2, 4, 1]

第 14 题 执行下面Python代码后，文件ip.txt中的内容用记事本打开时显示为？（ ）

```
1 with open("ip.txt", "w") as f:  
2     lst = ["202.206.224.21\n", "192.168.224.1\n"]  
3     f.writelines(lst)
```

A. 202.206.224.21192.168.224.1

B. 202.206.224.21
192.168.224.1

C. 202.206.224.21\n192.168.224.1

D. 202.206.224.21\n192.168.224.1\n

第 15 题

执行下面Python代码，输入0和字符串“abc”后，输出的结果是？（ ）

```

1  try:
2      a = int(input())
3      b = int(input())
4      x = a / b
5      print(x, end="#")
6  except ZeroDivisionError:
7      print(0, end="#")
8  except:
9      print(1, end="#")
10 else:
11     print(2, end="#")
12 finally:
13     print(3, end="#")

```

- A. 0#2#3#
- B. 0#3#
- C. 1#2#3#
- D. 1#3#

2 判断题（每题 2 分，共 20 分）

题号	1	2	3	4	5	6	7	8	9	10
答案	√	√	√	×	√	×	×	×	√	√

第 1 题 GESP测试是对认证者的编程能力进行等级认证，同一级别的能力基本上与编程语言无关。（ ）

第 2 题 在Python中，`print(list("GESP"))`执行后输出['G', 'E', 'S', 'P']。（ ）

第 3 题 执行下面Python代码后，输出两个True。

```

1  print("h" in "hello")
2  print("he" in "hello")

```

第 4 题 执行下面Python代码后，输出两个True。

```

1  lst1 = [1, 2, 3]
2  lst2 = lst1[:]
3  print(lst1 == lst2, lst1 is lst2)

```

第 5 题 执行下面Python代码，调用函数fun可以得到一个元组类型的数据。

```

1  def fun(a, b):
2      return a // b, a % b
3
4
5  print(type(fun(8, 5)))

```

第 6 题 Python程序中，全局变量与局部变量不允许重名。

第 7 题 下面这段程序的时间复杂度为平方阶 $O(n^2)$ 。（ ）

```
1 def fun(n):
2     sum1 = 0
3     for i in range(n):
4         sum1 += i
```

第 8 题 Python 在处理程序异常时，异常处理结构能够发现程序段中的语法错误。

第 9 题 要打开一个已经存在的图片文件并读取但不修改数据，则打开模式应该设定为rb。

第 10 题 执行下面Python代码后，输出的结果为za。

```
1 print(max('az', 'za', key=lambda a: a[0]))
```

3 编程题（每题 25 分，共 50 分）

3.1 编程题 1

- 试题名称：黑白方块
- 时间限制：1.0 s
- 内存限制：512.0 MB

3.1.1 题面描述

小杨有一个 n 行 m 列的网格图，其中每个格子要么是白色，要么是黑色。

对于网格图中的一个子矩形，小杨认为它是平衡的当且仅当其中黑色格子与白色格子数量相同。

小杨想知道最大的平衡子矩形包含了多少个格子。

3.1.2 输入格式

第一行包含两个正整数 n, m ，含义如题面所示。

之后 n 行，每行一个长度为 m 的 01 串，代表网格图第 i 行格子的颜色，如果为 0，则对应格子为白色，否则为黑色。

3.1.3 输出格式

输出一个整数，代表最大的平衡子矩形包含格子的数量，如果不存在则输出 0。

3.1.4 样例1

```
1 4 5
2 00000
3 01111
4 00011
5 00011
```

```
1 16
```

3.1.5 样例解释

对于样例1, 假设 (i, j) 代表第 i 行第 j 列, 最大的平衡子矩形的四个顶点分别为 $(1, 2)$, $(1, 5)$, $(4, 2)$, $(4, 5)$ 。

3.1.6 数据范围

对于全部数据, 保证有 $1 \leq n, m \leq 10$ 。

3.1.7 参考程序

```
1 N = 55
2 w = [[0] * N for _ in range(N)]
3
4 def check(xa, ya, xb, yb):
5     a = [0, 0]
6     for i in range(xa, xb + 1):
7         for j in range(ya, yb + 1):
8             a[w[i][j]] += 1
9     return a[0] == a[1]
10
11 def main():
12     n, m = map(int, input().split())
13     for i in range(1, n + 1):
14         s = input()
15         for j in range(1, m + 1):
16             w[i][j] = int(s[j - 1])
17     ans = 0
18     for i in range(1, n + 1):
19         for j in range(1, m + 1):
20             for ii in range(i, n + 1):
21                 for jj in range(j, m + 1):
22                     if check(i, j, ii, jj):
23                         ans = max(ans, (ii - i + 1) * (jj - j + 1))
24     print(ans)
25
26 if __name__ == "__main__":
27     main()
```

3.2 编程题 2

- 试题名称: 宝箱
- 时间限制: 1.0 s
- 内存限制: 512.0 MB

3.2.1 题面描述

小杨发现了 n 个宝箱, 其中第 i 个宝箱的价值是 a_i 。

小杨可以选择一些宝箱放入背包并带走, 但是小杨的背包比较特殊, 假设小杨选择的宝箱中最大价值为 x , 最小价值为 y , 小杨需要保证 $x - y \leq k$, 否则小杨的背包会损坏。

小杨想知道背包不损坏的情况下, 自己能够带走宝箱的总价值最大是多少。

3.2.2 输入格式

第一行包含两个正整数 n, k ，含义如题面所示。

第二行包含 n 个正整数 a_1, a_2, \dots, a_n ，代表宝箱的价值。

3.2.3 输出格式

输出一个整数，代表带走宝箱的最大总价值。

3.2.4 样例1

```
1 | 5 1
2 | 1 2 3 1 2
```

```
1 | 7
```

3.2.5 样例解释

在背包不损坏的情况下，小杨可以拿走两个价值为 2 的宝箱和一个价值为 3 的宝箱。

3.2.6 数据范围

对于全部数据，保证有 $1 \leq n \leq 1000, 0 \leq k \leq 1000, 1 \leq a_i \leq 1000$ 。

3.2.7 参考程序

```
1 N = 1010
2 a = [0] * N
3
4 def main():
5     n, k = map(int, input().split())
6     aa = input().split()
7     for i in range(1, n + 1):
8         a[i] = int(aa[i-1])
9     a[1:n + 1] = sorted(a[1:n + 1])
10    ans = 0
11    for i in range(1, n + 1):
12        sum = 0
13        for j in range(i, 0, -1):
14            if a[i] - a[j] <= k:
15                sum += a[j]
16            else:
17                break
18        ans = max(ans, sum)
19    print(ans)
20
21 if __name__ == "__main__":
22     main()
```