

Python - 作业 1

沈宇昊

210162701031

兰州理工大学

March 28, 2023



Contents

1. 第二章

2. 第三章

(1) - (2)

work.py — solution

```
work.py
1 import math
2 import numpy as np
3 PI, sqrt = math.pi, math.sqrt
4
5 print("2.1.1答案\n"+str(sqrt((PI**2)+3))+'\n')
6 print("2.1.2答案\n"+str(np.log(PI**2*sqrt(13+np.e)))+'\n')
7 print("2.1.3答案\n"+str(np.arctan(math.log(PI+1, 333))+'\n'))
8
9 print("2.2.1答案\n"+str(4/3*(PI**3))+'\n')
10 r1, r2 = 16.2, 9.4
11 print("2.2.2答案\n"+str(PI*((r1**2)-(r2**2)))+'\n')
12 r, h = 66, 24.2
13 print("2.2.3答案\n体积 "+str((r**2)*PI*h)+" 表面积 "+str(2*PI*r*(r+h))+'\n')
```

/opt/homebrew/bin/python3 /Users/syh/solution/work.py
 2.1.1答案
 3.5874238064353533
 2.1.2答案
 3.21526939770108
 2.1.3答案
 8.9126653809446899
 2.2.1答案
 41.341702448399755
 2.2.2答案
 546.8864491369112
 2.2.3答案
 体积 331171.61789669876 表面积 37405.658770701515

base system 00:23:50

base system 00:23:50

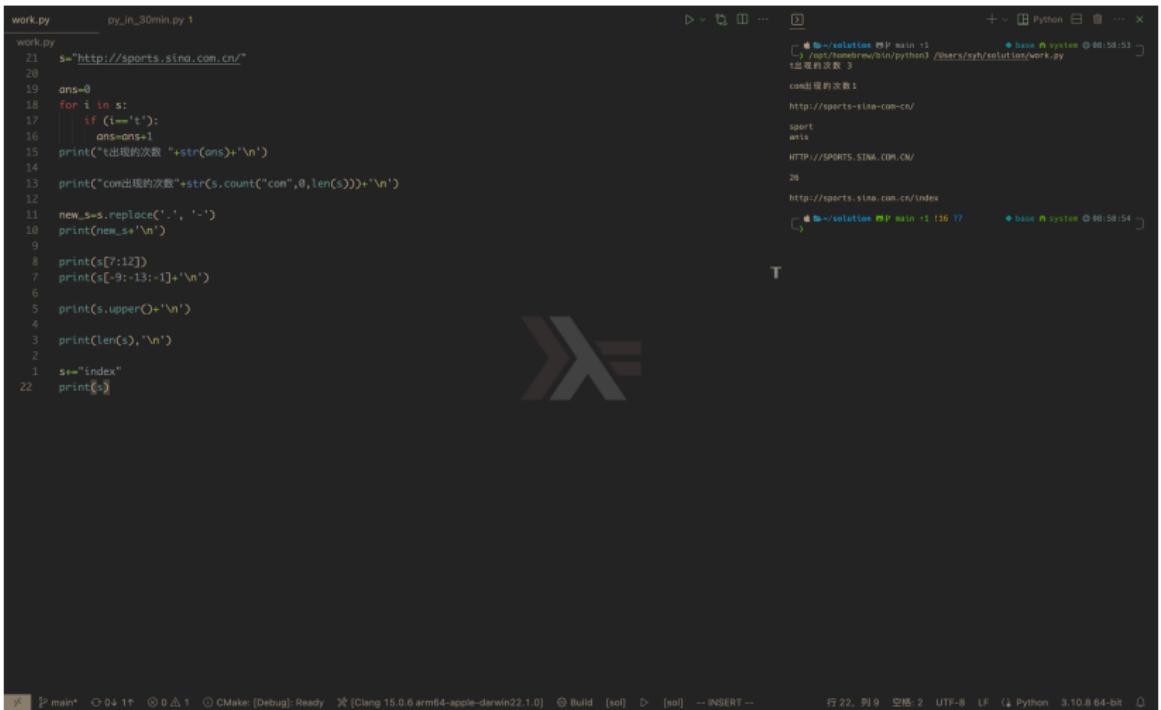
base system 00:23:50

(3)

The screenshot shows a Mac OS X desktop environment. In the center is a terminal window titled "work.py — solution". The terminal has two tabs open, both showing the command "python3 /Users/syh/solution/work.py" and its output: "1 1 2 1" and "1.4142135623738951". Below the terminal is a code editor window titled "work.py". The code is a Python script that takes four integers from input, calculates the distance between points (x1,y1) and (x2,y2), and prints the result rounded to two decimal places.

```
work.py
work.py
8 import math
9 n=input()
10 a=[]
11 for i in n.split():
12     a.append(int(i))
13 x1,y1,x2,y2=a[0],a[1],a[2],a[3]
14 ans=math.sqrt(((x1-x2)**2)+((y1-y2)**2))
15 ans=round(ans,2)
16 print(ans)
```

(4)



```

work.py          py_in_30min.py 1
work.py
21 s="http://sports.sina.com.cn/"
20
19 ans=0
18 for i in s:
17     if (i=='t'):
16         ans=ans+1
15 print("t出现的次数 "+str(ans)+"\n")
14
13 print("com出现的次数"+str(s.count("com",0,len(s)))+"\n")
12
11 new_s=s.replace('.','-')
10 print(new_s+"\n")
9
8 print(s[7:12])
7 print(s[-9:-13:-1]+'\n')
6
5 print(s.upper()+'\n')
4
3 print(len(s),'\n')
2
1 s+="index"
0 print(s)

```

The terminal window shows the execution of the Python script `work.py` and its output. The output includes the count of 't' and 'com' characters in the URL, the reversed string, and the length of the string plus an additional suffix.

(5) - (6)

```
work.py          py_in_30min.py 1
work.py
9  x1,x2,x3=input().split(' ')
8  ans=int(x1)*0.5+int(x2)*0.3+int(x3)*0.2
7  print("第5题答案",ans, '\n')
6
5  x=int(input())
4  ans=0
3  while(x>0):
2  |  ans+=x%10
1  |  x//=10
10 print("第六题",ans)
```



```
▶ ~/.solution └─ main :1      base ⚡ system ◉ 89:02:56
└─ 18 18 18
  └─ 第5题答案 18.0
    └─ 332
      └─ 第6题 :1
        └─ 116 17    4i  base ⚡ system ◉ 89:03:01
T
```

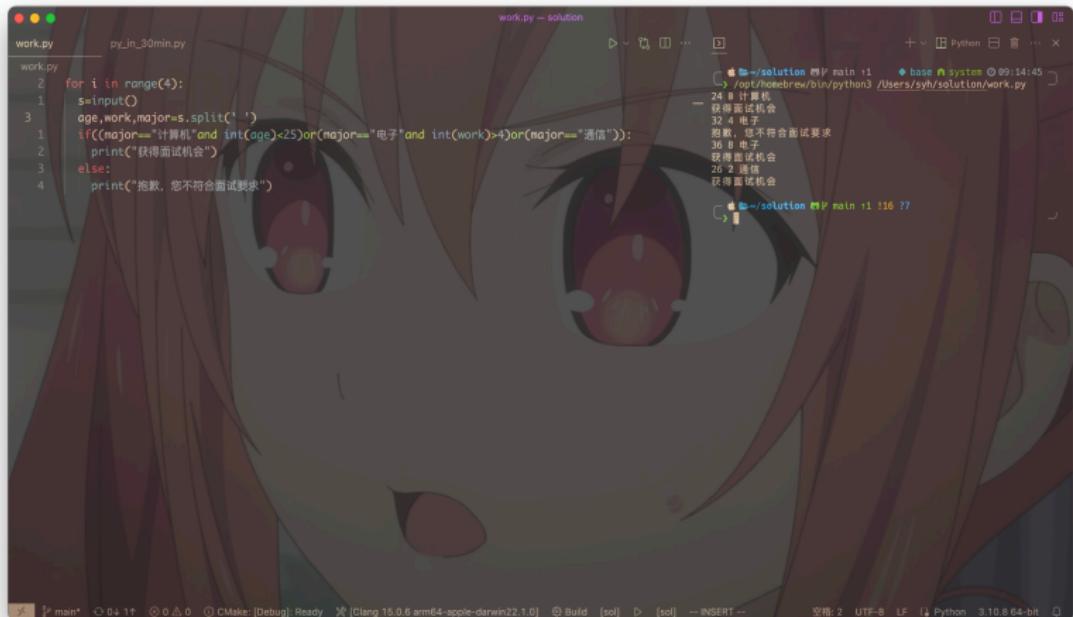
py main* ① 0 ④ 1 ⑤ 0 △ 1 ⑥ CMake [Debug] Ready ⑦ [Clang 15.0.0 arm64-apple-darwin22.1.0] ⑧ Build [sol] ⑨ [sol] --NORMA1 -- 行 10, 到 16 空格: 2 UTF-8 LF ↴ Python 3.10.8 64-bit ⑩

Contents

1. 第二章

2. 第三章

(1)



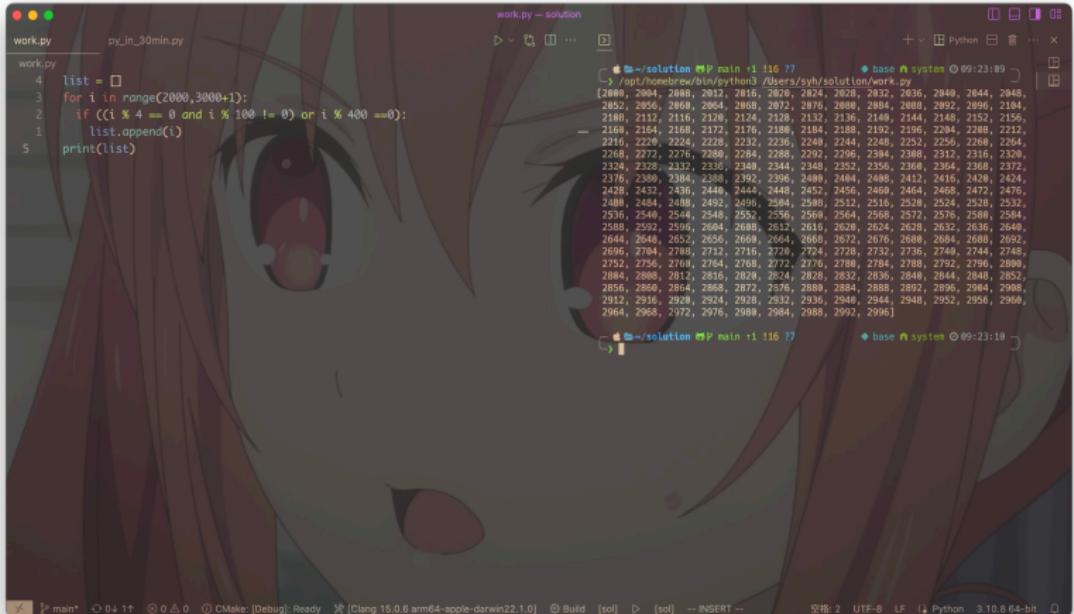
```
work.py -- solution
work.py      py_in_30min.py
1
2 for i in range(4):
3     s=input()
4     age,work,major=s.split(' ')
5     if((major=="计算机" and int(age)<25)or(major=="电子" and int(work)>4)or(major=="通信")):
6         print("获得面试机会")
7     else:
8         print("抱歉，您不符合面试要求")
```

```
重跑~/solution $ python main.py
24 8 计算机
32 4 电子
36 8 通信
36 8 电子
36 8 通信
26 2 通信
26 2 通信
26 2 通信
```

(2)

The image is a screenshot of a Mac OS X desktop. In the center is a terminal window titled 'work.py - solution' with the command '/opt/homebrew/bin/python3 /Users/syh/solution/work.py' entered. The output '成绩有误' is displayed. To the left of the terminal is a code editor showing the Python script 'work.py'. The script contains a series of nested if-elif statements for grading. The desktop background is a large, detailed anime-style illustration of a character's eye.

(3)



work.py py_in_30min.py

```

work.py
py_in_30min.py
1 list = []
2 for i in range(2000,3000+1):
3     if ((i % 4 == 0 and i % 100 != 0) or i % 400 == 0):
4         list.append(i)
5 print(list)

```

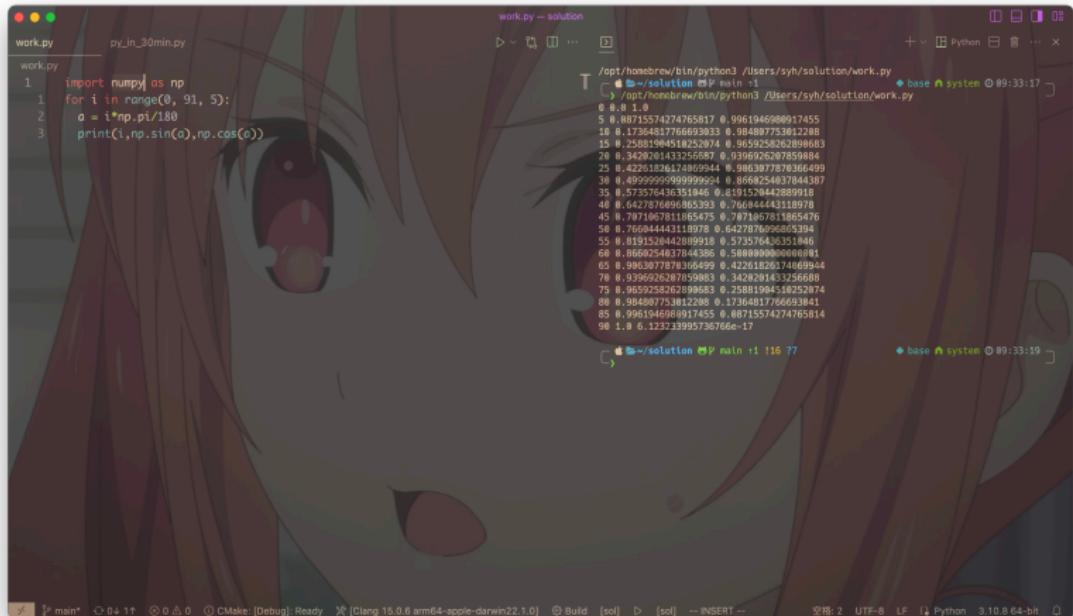
work.py — solution

```

base system 09:23:09
/home/syh/.local/bin/python3 /Users/syh/solution/work.py
[2080, 2084, 2088, 2013, 2816, 2826, 2824, 2028, 2832, 2836, 2940, 2844, 2948,
2852, 2856, 2868, 2064, 2868, 2872, 2876, 2068, 2884, 2888, 2892, 2890, 2194,
2198, 2112, 2116, 2120, 2124, 2128, 2132, 2136, 2140, 2144, 2148, 2152, 2156,
2160, 2164, 2168, 2172, 2176, 2186, 2184, 2188, 2192, 2196, 2294, 2288, 2212,
2216, 2220, 2224, 2228, 2232, 2236, 2248, 2244, 2248, 2252, 2256, 2268, 2264,
2268, 2272, 2276, 2280, 2284, 2288, 2292, 2296, 2304, 2308, 2312, 2316, 2320,
2324, 2328, 2332, 2336, 2340, 2344, 2348, 2352, 2356, 2360, 2364, 2368, 2372,
2376, 2380, 2384, 2388, 2392, 2396, 2400, 2404, 2408, 2412, 2416, 2420, 2424,
2428, 2432, 2436, 2440, 2444, 2448, 2452, 2456, 2460, 2464, 2468, 2472, 2476,
2480, 2484, 2488, 2492, 2496, 2504, 2508, 2512, 2516, 2520, 2524, 2528, 2532,
2536, 2540, 2544, 2548, 2552, 2556, 2560, 2564, 2568, 2572, 2576, 2580, 2584,
2588, 2592, 2596, 2598, 2608, 2612, 2616, 2620, 2624, 2628, 2632, 2636, 2640,
2644, 2648, 2652, 2656, 2668, 2664, 2666, 2672, 2676, 2688, 2684, 2688, 2692,
2696, 2704, 2708, 2712, 2716, 2720, 2724, 2728, 2732, 2736, 2740, 2744, 2748,
2752, 2756, 2760, 2764, 2768, 2772, 2776, 2780, 2784, 2788, 2792, 2796, 2800,
2804, 2808, 2812, 2816, 2820, 2824, 2828, 2832, 2836, 2840, 2844, 2848, 2852,
2856, 2860, 2864, 2868, 2872, 2876, 2880, 2884, 2888, 2892, 2896, 2894, 2983,
2912, 2916, 2920, 2924, 2928, 2932, 2936, 2940, 2944, 2948, 2952, 2956, 2960,
2964, 2968, 2972, 2976, 2980, 2984, 2988, 2992, 2996]
```

base system 09:23:10

(4)



work.py py_in_30min.py

```
work.py — solution
/opt/homebrew/bin/python3 /Users/syh/solution/work.py
└─ main.py
    └─ main
        └─ base
            └─ system
                └─ 09:33:17
                    └─ 1
                        └─ 0
                            └─ 1
                                └─ 0
                                    └─ 1
                                        └─ 0
                                            └─ 1
                                                └─ 0
                                                    └─ 1
                                                        └─ 0
                                                            └─ 1
                                                                └─ 0
                                                                    └─ 1
                                                                        └─ 0
                                                                            └─ 1
                                                                                └─ 0
                                                                                    └─ 1
                                                                                        └─ 0
                                                                                            └─ 1
                                                                                                └─ 0
                                                                                                    └─ 1
                                                                                                        └─ 0
                                                                                                            └─ 1
                                                                                                                └─ 0
                                                                                                                    └─ 1
................................................................
```

(5)



work.py

```
work.py
13 import math
14 def newton(m):
15     x0 = m/2 # 初始点, 也可以是别的值
16     x1 = x0/2 + m/(x0**2)
17     while abs(x1-x0) > 1e-5:
18         x0 = x1
19         x1 = x0/2 + m/(x0**2)
20     return x1
21
22 def main():
23     n=int(input())
24     for i in range(1, n+1):
25         _newton, _sqrt=newton(i), math.sqrt(i)
26         print(_newton, _sqrt)
27
28 main()
```

work.py — solution

```
work.py
13 import math
14 def newton(m):
15     x0 = m/2 # 初始点, 也可以是别的值
16     x1 = x0/2 + m/(x0**2)
17     while abs(x1-x0) > 1e-5:
18         x0 = x1
19         x1 = x0/2 + m/(x0**2)
20     return x1
21
22 def main():
23     n=int(input())
24     for i in range(1, n+1):
25         _newton, _sqrt=newton(i), math.sqrt(i)
26         print(_newton, _sqrt)
27
28 main()
```

Python 3.10.5

```
1.0
1.4142135623746899 1.4142135623730951
1.732050807568872 1.732050807568872
2.0 2.0
2.23606797749979 2.23606797749979
2.449489742783178 2.449489742783178
2.6457513110646933 2.6457513110645987
2.8284271247493797 2.8284271247493797
3.0 3.0
3.162277660168379 3.1622776601683795
```

base system 14:57:03

base system 14:57:09

base system 14:57:09

(6)

```
work.py -- solution
work.py
1 n=int(input())
2 print(n//17*17 if(n//17*17>0) else "不存在")
```

```
16
17
15
不存在
```

(7)

The screenshot shows a macOS desktop environment. In the center is a terminal window titled "work.py -- solution" with four tabs open, each showing the same Python code and its execution output:

```
work.py
work.py
4 n=int(input())
3 S=0
2 for i in range(1,n+1,2):
1   S+=1/i**((-1)**((i//2)%2+1))
5 S+=2
1 print(S)

/opt/homebrew/bin/python3 /Users/syh/solution/work.py
base system 18:00:07
> /opt/homebrew/bin/python3 /Users/syh/solution/work.py
1
1.0

base system 18:00:10
> /opt/homebrew/bin/python3 /Users/syh/solution/work.py
3
1.3333333333333333

base system 18:00:10
> /opt/homebrew/bin/python3 /Users/syh/solution/work.py
5
1.1333333333333333

base system 18:00:12
> /opt/homebrew/bin/python3 /Users/syh/solution/work.py
7
1.1433333333333333
```

Below the terminal is a code editor window titled "work.py" containing the following Python code:

```
work.py
work.py
4 n=int(input())
3 S=0
2 for i in range(1,n+1,2):
1   S+=1/i**((-1)**((i//2)%2+1))
5 S+=2
1 print(S)
```

The status bar at the bottom of the screen displays the following information:

- File: main*
- Editor mode: Insert
- Build status: CMake: [Debug]: Ready
- Compiler: Clang 15.0.6 arm64-apple-darwin22.1.0
- Build tools: Build [sol] [sol]
- Version: 3.10.8 64-bit

(8)



```
work.py — solution
```

work.py

```
1 work.py
 2 import random
 3 import math
 4 a, b, c = random.randint(1, 100), random.randint(
 5     1, 100), random.randint(1, 100)
 6 print(a,b,c)
 7 print(math.gcd(a,b,c), math.lcm(a,b,c))
```

Output:

```
1 $ cd ~/solution & python main.py
 2 $ cd ~/solution & python main.py
 3 $ /opt/homebrew/bin/python3 ./Users/syh/solutio
 4 n/work.py
 5 76 15 38
 6 1 1148
 7
 8 $ cd ~/solution & python main.py
 9 $ cd ~/solution & python main.py
10 $ /opt/homebrew/bin/python3 ./Users/syh/solutio
11 n/work.py
12 51 92 34
13 1 4692
14
15 $ cd ~/solution & python main.py
16 $ cd ~/solution & python main.py
17 $ /opt/homebrew/bin/python3 ./Users/syh/solutio
18 n/work.py
19 62 88 96
20 2 14888
21
22 $ cd ~/solution & python main.py
23 $ cd ~/solution & python main.py
24 $ /opt/homebrew/bin/python3 ./Users/syh/solutio
25 n/work.py
26
```