

期末模拟试卷-2

一、 单选题 （共 15 题， 30 分）

1、 该表达式的求值结果是什么？

```
expr = (7/2 == 7//2)
```

(2.0)

- A、 True
- B、 False
- C、 抛出异常
- D、 3

正确答案： B

解析：

2、 该表达式的求值结果是什么？

```
expr = not True and False
```

(2.0)

- A、 True
- B、 False
- C、 None
- D、 抛出异常

正确答案： B

解析：

3、 该表达式的求值结果是什么？

```
expr = -3**2
```

(2.0)

- A、 9
- B、 -9
- C、 SyntaxError 异常
- D、 6

正确答案： B

解析：

4、 以下关于 Python 变量命名的说法正确的是？

(2.0)

- A、 变量名可以以数字开头
- B、 变量名不区分大小写
- C、 Python 变量名区分大小写
- D、 变量名中不能包含下划线

正确答案： C

解析：

5、 以下表达式的求值结果是什么？

```
expr = int('1F', 16)
```

(2.0)

- A、 31
- B、 1F
- C、 15
- D、 抛出 ValueError 异常

正确答案： A

解析：

6、 下列代码执行后 result 的值是什么？

```
nums = [1, 2, 3]
```

```
chars = ['a', 'b', 'c']
```

```
result = list(zip(nums, chars))
```

(2.0)

- A、 [(1, 'a'), (2, 'b'), (3, 'c')]
- B、 [('1','a'), ('2','b'), ('3','c')]
- C、 ['1a', '2b', '3c']
- D、 zip 对象无法直接转换为列表，会报错

正确答案： A

解析：

7、下列代码执行后，打印结果是什么？

```
print("A", "B", "C", sep="-")
```

(2.0)

- A、 "A B C"
- B、 "A-B-C"
- C、 "A\tB\tC"
- D、 "ABC"

正确答案： B

解析：

8、以下代码执行后 lst 的值是什么？

```
lst = [5, 10, 15, 20]
```

```
lst.remove(10)
```

(2.0)

- A、 [5, 15, 20]
- B、 [5, 10, 15, 20, 10]
- C、 [5, 15, 20, 10]
- D、 抛出 ValueError 异常

正确答案： A

解析：

9、`lst = [1, 2]`

`res = lst * 3`

执行后 `res` 的值是什么？

(2.0)

A、`[1, 2, 1, 2, 1, 2]`

B、`[1, 2, [1, 2], [1, 2]]`

C、`[1, 2, 1]`

D、`[1, 2]*3` 报错

正确答案： A

解析：

10、`lst = [1, 2, 3]`

`x = lst[5:10]`

执行后 `x` 的值是什么？

(2.0)

A、抛出 `IndexError` 异常

B、`[3]`

C、`[]`

D、`[1, 2, 3, None, None]`

正确答案： C

解析：

11、`t = ((1,2), (3,4))`

`res = (1 in t)`

执行后 `res` 的值是什么？

(2.0)

A、 `True`

- B、 False
- C、 TypeError 异常
- D、 SyntaxError 异常

正确答案： B

解析：

12、执行后 duplicate 的值是什么？

```
original = {1, 2}
```

```
duplicate = original.copy()
```

```
original.add(3)
```

(2.0)

- A、 {1, 2}
- B、 {1, 2, 3}
- C、 空集 set()
- D、 抛出 TypeError 异常

正确答案： A

解析：

13、d = {'x': 10, 'y': 20}

```
val = d.get('z', 0)
```

执行后 val 的值是什么？

(2.0)

- A、 0
- B、 10
- C、

KeyError 异常

- D、 None

正确答案： A

解析：

```
14、nums1 = [1, 2, 3]
```

```
nums2 = [10, 20, 30]
```

```
res = map(lambda x, y: x + y, nums1, nums2)
```

将 res 转换为列表后，结果是什么？

(2.0)

A、 [11, 22, 33]

B、 [(1,10), (2,20), (3,30)]

C、 [1,2,3,10,20,30]

D、 抛出 TypeError 异常

正确答案： A

解析：

15、下列哪一种异常类型用于表示索引超出序列范围？

(2.0)

A、 IndexError

B、 KeyError

C、 TypeError

D、 AttributeError

正确答案： A

解析：

二、 填空题 （共 5 题， 20 分）

16、代码如下，执行结果如何？

```
x = -2
```

```
if x > 0:
```

```
    s = "Positive"
```

```
elif x == 0:
```

```
s = "Zero"
else:
    s = "Negative"

if 'g' in s:
    c = s.count('g')
else:
    c = len(s)

print(s)
print(c)
```

(4.0)

正确答案:

第 1 空:

Negative

第 2 空:

1

解析:

17、代码如下，请问执行后打印的两个结果分别是什么？

```
product = 1
for x in [2, 3, 5]:
```

```
        product *= x
length = 0
for y in "ab":
    length += 1
print(product)
print(length)
```

(4.0)

正确答案:

第 1 空:

30

第 2 空:

2

解析:

18、代码如下，请问执行后打印的两个结果分别是什么？

```
start = 5
acc = 0
while start <= 8:
    acc = acc + start*2
    start += 2
print(acc)
print(start)
```


(4.0)

正确答案:

第 1 空:

24

第 2 空:

9

解析:

19、代码如下，请问执行后打印的两个结果分别是什么？

```
acc = 0
skip_count = 0
for val in [1,2,3,4,5]:
    if val % 2 == 0:
        continue
    acc += val
    skip_count += 1
print(acc)
print(skip_count)
```

(4.0)

正确答案:

第 1 空:

9

第 2 空:

3

解析:

20、代码如下，请问执行后打印的两个结果分别是什么？

```
import copy  
lst = [5, [6,7]]  
lst_deep = copy.deepcopy(lst)  
lst_deep[1][1] = 700  
print(lst[1])  
print(lst_deep[1])
```

(4.0)

正确答案:

第 1 空:

[6, 7]

第 2 空:

[6, 700]

解析:

三、 判断题 （共 10 题， 20 分）

21、在同一个 Python 模块中定义两个同名的函数是合法的，后定义的函数会覆盖先定义的函数，不会产生语法错误。

(2.0)

正确答案： 正确

解析：

22、ValueError 常在参数类型正确但取值不适合操作时引发，例如使用 `int("abc")` 将字符串转换为整数时会引发 ValueError。

(2.0)

正确答案： 正确

解析：

23、使用 `pip install package==1.0.0` 命令可以安装指定版本号为 1.0.0 的第三方包。

(2.0)

正确答案： 正确

解析：

24、在使用 `readline()` 方法读取文件时，每次调用该方法都会返回文件的下一行，直到文件末尾。

(2.0)

正确答案： 正确

解析：

25、在 Python 中，函数定义中的非默认参数必须放在所有默认参数之后，否则会导致语法错误。

(2.0)

正确答案： 错误

解析：

26、在 Python 中，使用相对路径访问文件时，如果当前工作目录发生变化，文件访问路径不会变化。

(2.0)

正确答案： 错误

解析：

27、在 Python 中，循环体内部定义的变量在循环结束后仍然保留在全局命名空间中。

(2.0)

正确答案： 错误

解析：

28、eval 函数在 Python 中可以计算字符串表达式的值，并且只能返回表达式的结果，无法执行复杂的语句如循环或条件判断。

(2.0)

正确答案： 正确

解析：

29、在 f-字符串中，花括号内不能包含任何 Python 表达式，只能放置变量名。

(2.0)

正确答案： 错误

解析：

30、浮点数在 Python 中遵循 IEEE 754 标准，这导致某些十进制小数无法被精确表示为二进制浮点数。

(2.0)

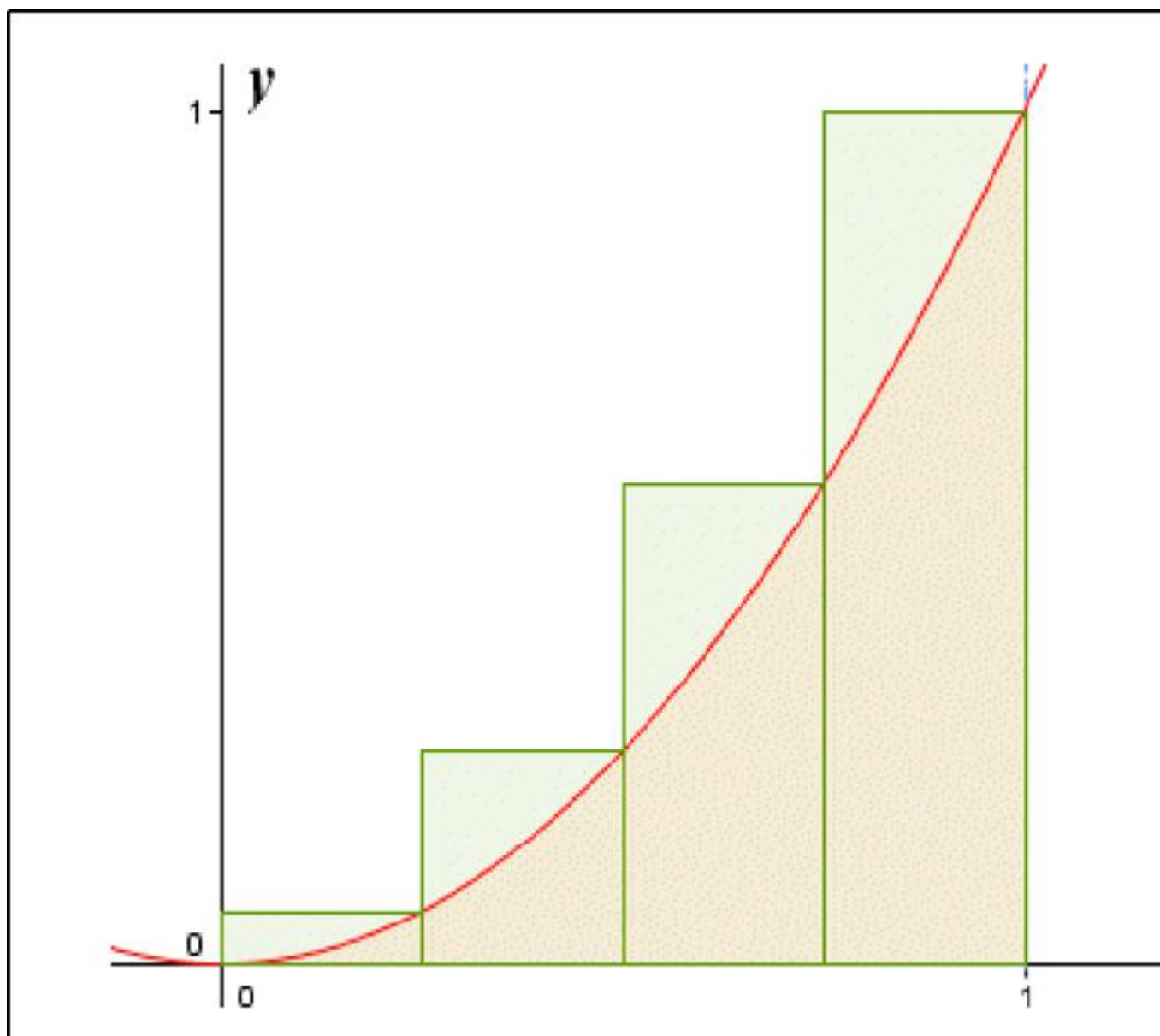
正确答案： 正确

解析：

四、 简答题 （共 3 题，30 分）

31、使用数值积分计算定积分（10 分）

计算定积分 $\int_0^1 x^2 dx$ 的值。已知该定积分的理论值为 1/3。请编写 Python 程序，使用数值积分（微元法）求解，具体要求如下：



1. 将积分区间 $[0,1]$ 等分为 N 份，使用右端点法计算各小矩形面积；（3 分）
2. 通过循环结构实现小矩形面积的累加求和；（3 分）
3. 实现以下功能：（4 分）
 - 设置 $N=1000000$ 以保证计算精度；
 - 计算数值解与理论值 $(1/3)$ 的相对误差(保留 8 位小数)；
 - 使用格式化输出显示数值解和相对误差；

(10.0)

正确答案:

```
import math

def calculate_integral(N):
    # 初始化累加和
    sum_of_area = 0

    # 计算每个小矩形宽度
    dx = 1.0 / N

    # 使用循环累加每个小矩形的面积
    for i in range(1, N + 1):
        # 右端点 x 坐标
        x = i * dx
        # 计算函数值  $f(x) = x^2$ 
        fx = x * x
        # 计算小矩形面积并累加
        sum_of_area += dx * fx

    # 计算理论值
    theoretical = 1/3

    # 计算相对误差(%)
    relative_error = abs(sum_of_area - theoretical) / theoretical * 100

    # 格式化输出结果
    print(f"数值积分结果: {sum_of_area:.8f}")
    print(f"理论值: {theoretical:.8f}")
    print(f"相对误差: {relative_error:.8f}%")

    return sum_of_area, relative_error

# 设定划分区间数
N = 1_000_000
result, error = calculate_integral(N)
```

解析:

[\[微元法题目解析.ipynb\]](#)

32、阶梯电价计算器（10 分）

设计程序计算家庭月度用电费用。具体要求：

1. 实现阶梯电价计算（5 分）：

0-180 度：0.5 元/度

181-450 度：0.55 元/度

451-750 度：0.75 元/度

750 度以上：1.0 元/度

2. 程序功能（5 分）：

输入月度用电量

计算各档位用电费用

输出详细计算过程和总费用

（10.0）

正确答案：

```
def calculate_electricity_bill(usage):  
    """  
    计算月度电费  
    参数: usage - 月度用电量（度）  
    返回: 阶梯计算明细和总费用  
    """  
    # 定义阶梯电价表: (起始用量, 结束用量, 单价)  
    price_levels = [  
        (0, 180, 0.50),      # 第一档: 0-180 度  
        (180, 450, 0.55),   # 第二档: 181-450 度  
        (450, 750, 0.75),   # 第三档: 451-750 度  
        (750, float('inf'), 1.00) # 第四档: 750 度以上  
    ]  
  
    total_fee = 0
```

```

bill_details = []
remaining_usage = usage

# 逐级计算电费
for start, end, rate in price_levels:
    if remaining_usage <= 0:
        break

    # 计算当前档位的用电量
    level_usage = min(remaining_usage, end - start)
    if level_usage > 0:
        level_fee = level_usage * rate
        total_fee += level_fee

    # 记录计算明细
    bill_details.append({
        'range': f"{start}-{end if end != float('inf') else '以上'
' }",
        'usage': level_usage,
        'rate': rate,
        'fee': level_fee
    })

    remaining_usage -= level_usage

return bill_details, total_fee

def print_electricity_bill(usage):
    """生成电费账单报告"""
    print("\n" + "="*40)
    print("月度电费账单")
    print("="*40)

    print(f"本月用电总量: {usage:,.1f}度")

    # 获取计算结果
    details, total = calculate_electricity_bill(usage)

    # 打印分段计算明细
    print("\n 阶梯计算明细: ")
    for detail in details:
        print("\n" + "-"*35)
        print(f"用电区间: {detail['range']}度")
        print(f"用电量: {detail['usage']:.1f}度")
        print(f"单价: {detail['rate']:.2f}元/度")
        print(f"金额: {detail['fee']:.2f}元")

    # 打印汇总信息
    print("\n" + "="*40)
    print(f"应缴电费: {total:.2f}元")
    print(f"平均电价: {(total/usage):.3f}元/度")
    print("="*40)

# 测试不同用电量的情况
test_cases = [
    120,      # 仅第一档

```



```
    300,    # 涉及前两档
    600,    # 涉及前三档
    900     # 涉及所有档位
]

for usage in test_cases:
    print_electricity_bill(usage)
```

解析：

[\[阶梯电量题目解析.ipynb\]](#)

33、月度销售报告（10 分）

企业需要定期生成月度销售报告。假设有一个名为 `monthly_sales.txt` 的文本文件，文件中每一行记录一个月的销售数据，格式如下：

月份,销售总额,成本总额

例如：

2024-01,150000,90000

2024-02,200000,120000

2024-03,180000,110000

请编写一个程序，读取 `monthly_sales.txt` 文件，计算每个月的利润率，并将结果写入一个名为 `monthly_profit_margin.txt` 的文件中。利润率的计算公式为：

$$\text{利润率} = \left(\frac{\text{销售总额} - \text{成本总额}}{\text{销售总额}} \right) \times 100\%$$

输出文件的格式应为：

月份：利润率%

例如：

2024-01: 40.0%

2024-02: 40.0%

2024-03: 38.9%

评分点：

- 正确读取并解析文本文件内容（3 分）
- 正确计算每个月的利润率，保留一位小数（4 分）
- 正确将结果写入输出文件，格式符合要求（3 分）

(10.0)

正确答案：

```
def generate_profit_margin_report(input_file, output_file):
    with open(input_file, 'r', encoding='utf-8') as f:
        lines = f.readlines()

    with open(output_file, 'w', encoding='utf-8') as f_out:
        for line in lines:
            line = line.strip()
            if not line:
                continue # 跳过空行
            parts = line.split(',')
            month = parts[0]
            profit_margin = float(parts[1])
            f_out.write(f"{month}: {profit_margin:.1f}%\n")
```

```
if len(parts) != 3:
    continue # 跳过格式错误的行
month, revenue, cost = parts
try:
    revenue = float(revenue)
    cost = float(cost)
except ValueError:
    f_out.write(f"{month}: 利润率无法计算\n")
    continue
if revenue == 0:
    f_out.write(f"{month}: 利润率无法计算\n")
    continue
profit_margin = ((revenue - cost) / revenue) * 100
f_out.write(f"{month}: {profit_margin:.1f}%\n")

# 示例运行
generate_profit_margin_report('monthly_sales.txt',
                              'monthly_profit_margin.txt')
```

解析:

[\[月度销售报告题目解析.ipynb\]](#)