

北京工业大学 2020——2021 学年第 1 学期

《Python 编程基础》 考试试卷 A 卷

考试说明： 考试时间：95 分钟；考试形式：闭卷

适用专业：机器人工程

承诺：

本人已学习了《北京工业大学考场规则》和《北京工业大学学生违纪处分条例》，承诺在考试过程中自觉遵守有关规定，服从监考教师管理，诚信考试，做到不违纪、不作弊、不替考。若有违反，愿接受相应的处分。

承诺人： 学号： 班号：

注：本试卷共 五 大题，共 12 页，满分 100 分，考试时必须使用卷后附加的统一答题纸或草稿纸。请将答案统一写在答题纸上，答案写在其他位置造成的成绩缺失由考生自己负责。

卷面成绩汇总表（阅卷教师填写）

题号	一	二	三	四	五	总成绩
满分	20	20	20	10	30	
得分						

得分

一、判断题（每题 2 分，共 20 分）

请在答题纸对应题号下方空格处为正确说法打√，错误说法打×。

1. 已知 $x = [3]$ ，那么执行语句 $x = x + [6]$ 之后， x 的内存地址不改变。（ ）
2. 正则表达式对象的 `match()` 方法可以在字符串的指定位置开始进行指定模式的匹配。（ ）
3. Python 变量使用前必须先声明，并且一旦声明仍然可在当前作用域内改变其类型。（ ）
4. 对于 $a = [1, 2]$ 和 $b = (1, 2)$ 两个数据对象而言， a 所占内存大于 b 所占的内存。（ ）
5. Python 支持多继承，如果不同父类中有相同的方法名，则 Python 解释器报错。（ ）

6. 字典属于 Python 有序序列, 和列表、字符串、元组一样都支持双向索引。()
7. 扩展库 os 中的方法 remove() 不可以删除带有只读属性的文件。()
8. 使用内置函数 open() 且以 "w" 模式打开的文件, 文件指针默认指向文件头。
()
9. Python 中的列表被切片后, 列表中的某个元素被删除后别的元素将不再自动收缩补齐空格。()
10. 语法 C = "ab" + "ab" 比 "ab".append("ab") 更快。()

得分

二、单选题 (每题 2 分, 共 20 分)

请在答题纸对应题号下方空格处填入 A、B、C、D 中的一个选项。

1. 关于 while 循环和 for 循环的区别, 下列说法正确的是 ()。
- A. while 语句的循环体至少无条件执行一次, for 语句的循环体有可能一次都不执行
- B. while 语句智能由于循环次数未知的循环, for 语句只能用于循环次数已知的循环
- C. 在很多情况下, while 语句和 for 语句可以等价使用
- D. while 语句只能用于可迭代变量, for 语句可以用任意表达式表示条件
2. 设 str = 'python', 想把字符串的最后一个字母大写, 其他字母还是小写, 正确的选项是 ()。
- A. print(str[end].upper()+str[1:]) B. print(str[6].upper()+str[-1:1])
- C. print(str[5].upper()+str[1:-1]) D. print(str[0:-1]+str[5].upper())
3. 以下关于列表操作的描述, 错误的是 ()。
- A. 通过 append 方法可以向列表添加元素
- B. 通过 extend 方法可以将另一个列表中的元素逐一添加到列表中
- C. 通过 insert(index,object) 方法在指定位置 index 前插入元素 object
- D. 通过 add 方法可以向列表添加元素
4. 以下关于元组操作的描述, 正确的是 ()。
- A. 通过 append 方法可以向列表添加元素
- B. 通过 extend 方法可以将另一个列表中的元素逐一添加到列表中

- C. 通过 `insert(index,object)` 方法在指定位置 `index` 前插入元素 `object`
- D. 通过内置函数 `list()` 可以将元组变为列表对象
5. 关于面向对象的继承，以下选项中描述错误的是（ ）。
- A. 继承是指一组对象所具有的部分相似性质
- B. 继承是指类之间共享属性和操作的机制
- C. 继承是指各类之间的共同性质
- D. 继承是指一个类具有另一个类的性质
6. 关于 Python 内存管理，下列说法错误的是（ ）。
- A、变量不必事先声明 B、变量无须先创建和赋值而直接使用
- C、变量无须指定类型 D、可以使用 `del` 释放资源
7. 关于以下程序输出的两个值的描述正确的是（ ）。
- ```
da = [1,2,3]
print(id(da))
def getda(st):
 fa = da.copy()
 print(id(fa))
getda(da)
```
- A. 两个值相等                                  B. 每次执行的结果不确定
- C. 首次不相等                                  D. 两个值不相等
8. 对于列表 `ls` 的操作，以下选项中描述错误的是（ ）。
- A. `ls.clear()`: 删除 `ls` 的最后一个元素
- B. `ls.copy()`: 生成一个新列表，复制 `ls` 的所有元素
- C. `ls.reverse()`: 列表 `ls` 的所有元素反转
- D. `ls.append(x)`: 在 `ls` 最后增加一个元素
9. 现有代码 `t=('a')`，在 `python3` 解释器中查看 `type(t)` 得到的结果为（ ）。
- A    `<class 'str'>`                                  B    `<class 'tuple'>`
- C    `(class 'str')`                                  D    `(class 'tuple')`
10. 以下不能创建一个字典的语句是（ ）
- A    `dict1 = {}`                                          B、 `dict2 = { 3 : 5 }`

C、dict3 = dict( [2 , 5] ,[ 3 , 4 ] )

D、dict4 = dict( ( [1,2],[3,4] ) )

|     |
|-----|
| 得 分 |
|     |

## 三、填空题（每空 2 分，共 20 分）

请在答题纸对应题号后的空格处顺序填写答案。

1. 已知列表  $x = [(1), 2]$ ，执行语句  $y = x[1]$  后，表达式  $\text{id}(x[1]) == \text{id}(y)$  的值为\_\_\_\_\_。
2. 表达式  $(i \text{ for } i \text{ in range}(10) \text{ if } i > 2)$  的值为\_\_\_\_\_。
3. 已知  $x = [1, 2, 1]$ ，那么表达式  $\text{id}(x[0]) == \text{id}(x[2])$  的值为\_\_\_\_\_。
4. 正则表达式模块 `re` 的\_\_\_\_\_方法用来在整个字符串中进行指定模式的匹配。
5. 表达式 `eval(""" import( 'math' ).sqrt(3**2+4**2) """)` 的值为\_\_\_\_\_。
6. 使用列表推导式得到 100 以内所有能被 17 整除的数的代码可以写作\_\_\_\_\_。
7. 表达式 `list(filter(lambda x:x>3, [0,1,2,3,0,0]))` 的值为\_\_\_\_\_。
8. 假设有 Python 程序文件 `demo.py`，代码如下：

```
def main():
```

```
 if __name__ == '__main__':
```

```
 print(1)
```

```
 else:
```

```
 print(2)
```

```
main()
```

将该程序文件直接运行时输出结果为\_\_\_\_\_，作为模块导入时得到结果\_\_\_\_\_。

9. 表达式 `list(map(lambda x: len(x), [ '[1, 2]', 'bb', 'ccc' ]))` 的值为\_\_\_\_\_。
10. 表达式 `len( 'hello world' [2:])` 的值为\_\_\_\_\_。

|     |
|-----|
| 得 分 |
|     |

## 四、程序分析题（每空 2 分，共 10 分）

请在答题纸对应题号后的空格处顺序填写答案。

1. 将函数补充完整，使其具有以下功能：给定一个整数数组 `nums` 和一个目标值 `target`，请你在该数组中找出差为目标值的两个整数，并返回他们的数组下

标。给定 `nums = [2, 7, 11, 15]`, `target = -9`, `nums[0] - nums[1] = 2 - 11 = -9`, 所以返回 `(0, 3)`。

```
def twoDiff(nums, target):
 for i in range(len(nums)):
 for j in range(i, len(nums)):

 return i, i+j
 return False
```

2. 下列程序的输出结果为：\_\_\_\_\_。

```
class Student(object):
 count = 0
 def __init__(self, name, age):
 self.name = name
 self.age = age
 if age > 18
 Student.count += 1
 @staticmethod
 def count_student():
 print(共实例%d 个对象' % Student.count)

bob = Student('bob', 19)
Jenny = Student('Jenny', 18)
Danny = Student('Danny', 19)
liming = Student('liming', 20)
Student.count_student()
```

3. 下列程序的输出结果是\_\_\_\_\_。

```
def generator_example():
 yield 1
 yield 2
if __name__ == '__main__':
 for e in generator_example():
 print(e)
```

4. 下列程序表示计算员工工资，工资与工龄有关，计算公式为  $\text{salary} = 4000 + 200 * \text{age}$ ，输出更改前后的工资。现有一员工 jack，原始工龄为 4，现在工龄为 5， 请补充程序空缺：

```
class Worker:

 def __init__(self, name, workAge):

 self.name = name

 self.workAge = workAge

 def salary(self):

 return _____

 salary = property(salary)

 def dispSalary(self):

 print(self.salary)
```

```
jack = Worker('jack', 4)
```

```
jack.dispSalary()
```

```
jack.workAge = 5
```

```
jack.dispSalary()
```

11. 给出输出结果\_\_\_\_\_。

```
>>> L = []
>>> print(id(L))
>>> 259766
```

```
>>>L.append("I")
```

```
>>>print(id(L))
```

|     |
|-----|
| 得 分 |
|     |

### 五、编程题（共 30 分）

请在答题纸对应题号下方标明具体题号、顺序作答。

1. 冒泡法对列表 `list=[5,10,6,7,50,2,13,49]` 排序（10 分）。

2. 斐波那契数，通常用  $F(n)$  表示，形成的序列称为斐波那契数列。该数列由 0 和 1 开始，后面的每一项数字都是前面两项数字的和。也就是：

$$F(0) = 0, \quad F(1) = 1$$

$$F(N) = F(N - 1) + F(N - 2), \text{ 其中 } N > 1。$$

用函数形式表示  $F(N)$ 。（10 分）

3. 学校在拍年度纪念照时，一般要求学生按照 非递减 的高度顺序排列。请你返回能让所有学生以非递减高度排列的最小必要移动人数。（10 分）

**示例：**如果学生身高记录为 `heights`，而排列位置记为列表的索引

输入：`heights = [1,1,4,2,1,3]`（表示 6 为学生，身高分别为 1,1,4,2,1 和 3）

输出：3

解释：

当前学生的排列为：`[1,1,4,2,1,3]`

目标排列为：`[1,1,1,2,3,4]`，（索引）2、4 和 5 处的学生需要移动。

## 答 题 纸

姓名: \_\_\_\_\_ 学号: \_\_\_\_\_

|     |
|-----|
| 得 分 |
|     |

一、判断题（每题 2 分，共 20 分）

请在题号下方空格处为正确说法打√，错误说法打×。

|   |   |   |   |   |   |   |   |   |    |
|---|---|---|---|---|---|---|---|---|----|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|   |   |   |   |   |   |   |   |   |    |

|     |
|-----|
| 得 分 |
|     |

二、单选题（每题 2 分，共 20 分）

请在题号下方空格处填入 A、B、C、D 中的一个选项。

|   |   |   |   |   |   |   |   |   |    |
|---|---|---|---|---|---|---|---|---|----|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|   |   |   |   |   |   |   |   |   |    |

|     |
|-----|
| 得 分 |
|     |

三、填空题（每题 2 分，共 20 分）

请在题号后的空格处顺序填写答案。

1. \_\_\_\_\_
2. \_\_\_\_\_
3. \_\_\_\_\_
4. \_\_\_\_\_
5. \_\_\_\_\_
6. \_\_\_\_\_
7. \_\_\_\_\_ , \_\_\_\_\_
8. \_\_\_\_\_
9. \_\_\_\_\_
10. \_\_\_\_\_



|     |
|-----|
| 得 分 |
|     |

四、程序分析题（每空 2 分，共 10 分）

请在题号后的空格处顺序填写答案。

1. \_\_\_\_\_
2. \_\_\_\_\_
3. \_\_\_\_\_ , \_\_\_\_\_
4. \_\_\_\_\_

|     |
|-----|
| 得 分 |
|     |

五、编程题（每题 10 分，共 30 分）

请标明具体题号、按要求顺序作答。

1.

2.

**3.**

草 稿 纸

姓名： \_\_\_\_\_

学号： \_\_\_\_\_