**BÀI GIẢI MỘT SỐ BÀI TẬP CƠ BẢN**

Tuần 5: Chương 4: Mảng 2 chiều:

Bài tập 1: Cho 1 mảng 2 chiều. Hãy cài đặt các chức năng như sau:

* Nhập / xuất mảng 2 chiều
* Tính tổng mảng 2 chiều
* Tính tích mảng 2 chiều
* Tìm giá trị lớn nhất
* Tìm giá trị nhỏ nhất

Nhập xuất mảng 2 chiều

|  |
| --- |
| def nhap\_mang\_2c(m,n):      arr = []      for r in range(m):          b = []          for c in range(n):              x = int(input(f"Nhập vào phần tử {r+1}-{c+1}: "))              b.append(x)          arr.append(b)      return arr  def xuat\_mang\_2c(arr):      for r in arr:          for c in r:              print(c, end=" ")          print()  def tong\_mang(arr):      tong = 0      for r in arr:          for c in r:              tong += c      return tong  def tinh\_tich(arr):      tich = 1      for r in arr:          for c in r:              tich \*= c      return tich  def tim\_max(arr):      max = arr[0][0]      for r in arr:          for c in r:              if max < c:                  max = c      return max  def tim\_min(arr):      min = arr[0][0]      for r in arr:          for c in r:              if min > c:                  min = c      return min  if \_\_name\_\_ == "\_\_main\_\_":      m = int(input("Nhập vào số dòng: "))      n = int(input("Nhập vào số cột: "))      arr = nhap\_mang\_2c(m,n)      xuat\_mang\_2c(arr)      tong = tong\_mang(arr)      tich = tinh\_tich(arr)      max = tim\_max(arr)      min = tim\_min(arr)      print(f"Tổng các phần tử của mảng: {tong}")      print(f"Tích các phần tử của mảng: {tich}")      print(f"Giá trị lớn nhất của mảng: {max}")      print(f"Giá trị nhỏ nhất của mảng: {min}") |

**Tuần 6: Chương 5: Chuỗi**

**Bài 1**: Cho phép người dùng nhập 2 chuỗi s1, s2. Xuất ra màn hình kết quả tổng của 2 chuỗi đó

*Gợi ý: dùng hàm input yêu cầu người dùng nhập giá trị*

* *Nhập chuỗi thư nhất và gán vào biến s1*
* *Nhập chuỗi thứ hai và gán vào biến s2*
* *Xuất ra màn hình 2 chuổi trên hoặc gán 2 chuổi s1, s2 vào 1 chuỗi rồi xuất ra kết quả chuỗi cuối cùng*

**Bài 2**: Cho phép người dùng nhập vào một chuỗi. Đổi tất cả ra chữ thường/đổi tất cả ra chữ hoa và xuất ra kết quả ra màn hình

*Gợi ý: dùng hàm input yêu cầu người dùng nhập giá trị*

* *Nhập chuỗi thư nhất và gán vào biến s*
* *Dùng hàm lower đề đổi chuổi thành chữ thường*
* *Dùng hàm upper đề đổi chuổi thành chữ HOA*
* *Xuất ra màn hình 2 chuổi trên*

**Bài 3**: Viết hàm tìm xem chuỗi S1 có trong chuỗi S2 hay không? Ví dụ:

S1 = “viet nam dat nuoc toi yeu”;

S2 = “dat nuoc”;

Tìm xem chuỗi S2 có xuất hiện trong chuỗi S1 không?

*Gợi ý: dùng hàm input yêu cầu người dùng nhập giá trị*

* *Xem từ khóa in, not in (kiểm tra chuỗi trong chuỗi)*

Giải bài 1:

|  |
| --- |
| def cong\_chuoi(s1,s2):      return s1+" "+s2  if \_\_name\_\_ == "\_\_main\_\_":      s1 = input("Xin vui lòng nhập vào chuỗi s1: ")      s2 = input("Xin vui lòng nhập vào chuỗi s2: ")      kq = cong\_chuoi(s1,s2)      print(f"Tổng chuỗi vừa nhập là: {kq}") |

Giải bài 2:

|  |
| --- |
| def doi\_chuoi(string):      new\_string = ''      for char in string:          if char >= 'A' and char <= 'Z':              char = chr(ord(char) + 32)          elif char >= 'a' and char <= 'z':              char = chr(ord(char) - 32)          new\_string += char      return new\_string  if \_\_name\_\_ == "\_\_main\_\_":      s = input("Nhập vào đoạn chuỗi cần chuyển: ")      kq = doi\_chuoi(s)      print(f"Kết quả: {kq}") |

Giải bài 3:

|  |
| --- |
| def tim\_chuoi(s1,s2):      if s2 in s1:          return True      return False  if \_\_name\_\_ == "\_\_main\_\_":      string1= input("Nhập vào chuỗi s1: ")      string2= input("Nhập vào chuỗi s2: ")      kq = tim\_chuoi(string1,string2)      if kq:          print("Chuỗi s1 có trong chuỗi s2")      else:          print("Chuỗi s1 không có trong chuỗi s2") |

**Tuần 7: Chương 6: File**

**Bài 1**. Viết chương trình đọc nội dung của một tập tin văn bản cho trước. Tên tập tin do người dùng chỉ định. Trong tập tin có nhiều dòng chữ.

Ví dụ: Tên tập tin: abc.txt

Nội dung:

Hello World

I Love You Very Much

I am here. Where are you?

|  |
| --- |
| def doc\_file():      with open("abc.txt",mode='r',encoding='utf-8')as f:          s = f.read()          f.close()      return s  if \_\_name\_\_ == "\_\_main\_\_":      kq = doc\_file()      print(f"Đọc file \n{kq}") |

**Bài 2**. Viết chương trình đọc một danh sách các số được ghi trong một tập tin văn bản, với mỗi số cách nhau bằng dấu khoảng trắng. Hiển thị danh sách ra màn hình và tính tổng các số đó.

Ví dụ: Tên tập tin: abc.txt

Nội dung: 1 2 3 4 5.5 6.5

Tổng là: 22

|  |
| --- |
| def doc\_file():      with open("xyz.txt",mode='r',encoding='utf-8')as f:          s = f.read()          f.close()      return s  def tinh\_tong():      tong = 0      string = doc\_file()      a = string.split(" ")      for i in a:          tong = tong + float(i)      return tong  if \_\_name\_\_ == "\_\_main\_\_":      kq = tinh\_tong()      print(f"Tổng giá trị \n{kq}") |

**Bài 3**. Viết chương trình ghi thêm nội dung vào một tập tin văn bản đang có trên hệ thống.

Ví dụ: Tên tập tin: abc.txt

Nội dung đang có: “Hello”

Nội dung thêm: “my love”

|  |
| --- |
| def doc\_file(string):      with open("abc.txt",mode='a',encoding='utf-8')as f:          s = f.write(string)          f.close()      return s  if \_\_name\_\_ == "\_\_main\_\_":      string = input("Nhâp đoạn chuỗi cần thêm: ")      doc\_file(string) |

**Tuần 8: Chương 7: Dictionary, Tuple**

**1.Dictionary:**

**Bài 1**. Viết một đoạn lệnh cho phép người dùng thêm một khóa vào từ điển. Trong đó khóa và giá trị do người dùng chỉ định:

Ví dụ:

Từ điển trước khi thêm: {0: 10, 1: 20}

Phần tử cần thêm: key = 2, value = 30

Từ điển sau khi thêm (không cần đúng thứ tự như trong ví dụ):

{0: 10, 1: 20, 2: 30}

|  |
| --- |
| def nhap\_dl():      n = input("Nhập vào dữ liệu: ")      if n.isnumeric():          number = int(n)      else:          number = n      return number  def them\_pt(dict,key,value):      dict[key] = value      return dict  if \_\_name\_\_ == "\_\_main\_\_":      dictionary = {0: 10, 1: 20}      print("Nhập vào key.")      key = nhap\_dl()      print("Nhập vào value.")      value = nhap\_dl()      kq = them\_pt(dictionary,key,value)      print(f"Phần tử sau khi thêm {dictionary}") |

**Bài 2**. Viết đoạn lệnh để kiểm tra xem một khóa có nằm trong từ điển không. Nếu có, xuất ra “Tồn tại giá trị trong từ điển với giá trị là v” với v là giá trị có trong từ điển. Nếu không, xuất ra là “Không có khóa k trong từ điển” với k là khóa đã nhập.

Ví dụ: {“a”: 1, “b”: 2}

Tìm “a”

Tồn tại giá trị trong từ điển với giá trị là 1

|  |
| --- |
| def check\_dict(dict,key):      if key in dict:          return 1      return -1  def nhap\_dl():      n = input("Nhập vào dữ liệu: ")      if n.isnumeric():          number = int(n)      else:          number = n      return number  if \_\_name\_\_ == "\_\_main\_\_":      dictionary = {'a': 1, 'b': 2}      print("Nhập key cần tìm: ")      key = nhap\_dl()      kq = check\_dict(dictionary,key)      if kq == 1:          print(f"Phần tử key {key} cần tìm có trong từ điển là {dictionary[key]}")      else:          print(f"Không tìm thấy key {key} trong từ điển") |

**Bài 3**. Viết chương trình xóa một phần tử với khóa cho trước khỏi từ điển. Xuất lại từ điển sau khi xóa.

Ví dụ: {“a”: 1, “b”: 2, “c”: 3.5, “d”: “hello”}

Xóa phần tử với khóa: “c”

Từ điển sau khi xóa (không cần đúng thứ tự như ví dụ):

{“a”: 1, “b”: 2, “d”: “hello”}

|  |
| --- |
| def check\_dict(dict,key):      if key in dict:          del dict[key]          return dict      return -1  def nhap\_dl():      n = input("Nhập vào dữ liệu: ")      if n.isnumeric():          number = int(n)      else:          number = n      return number  if \_\_name\_\_ == "\_\_main\_\_":      dictionary = {'a': 1, 'b': 2, 'c': 3.5, 'd': 'hello'}      print("Nhập key cần xóa: ")      key = nhap\_dl()      kq = check\_dict(dictionary,key)      if kq:          print(f"Phần tử key {key} vừa xóa trong từ điển {dictionary}")      else:          print(f"Không tìm thấy key {key} cần xóa trong từ điển") |

2. **Tuple**

Nhập xuất 1 tuple trong Python

|  |
| --- |
| def nhap\_tuple(n):      arr = []      for i in range(n):          x = nhap\_dl()          arr.append(x)      return tuple(arr)  def nhap\_dl():      n = input("Nhập vào dữ liệu: ")      if n.isnumeric():          number = int(n)      else:          number = n      return number  if \_\_name\_\_ == "\_\_main\_\_":      n = int(input("Nhập số lượng phần tử: "))      kq = nhap\_tuple(n)      print(kq) |

**Bài 1**: Viết chương trình sinh một tuple chứa các phần tử có các kiểu dữ liệu khác nhau.

Sau đó, unpack các phần tử trong một tuple.

**Bài 2**: Viết chương trình chuyển một tuple sang thành list và ngược lại từ list sang tuple

**Bài 3**: Viết chương trình đảo ngược một tuple.