Computer Science Holiday Homework

(Session 2022-2023)

Student’s Name: Harsh Narayan Jha

Roll: 09

b Section - A b

1. “Welcome” is string literal.

2. $ symbol can be used in naming an identifier -> False

3. Write any 2 data types available in Python.

>>> string and list.

4. range(1, 10) will return values in the range of 1 to 10. (10 being exclusive!)

>>> In Mathematical notation, it will return values in the range of [1, 10)

5. randint(1, 10) will return values in the range of 1 to 10. (10 being inclusive!)

>>> In Mathematical notation, it will return values in the range of [1, 10]

6. Raj was working on application where he wanted to divide the two number (A and B), he has written the expression as C = A/B, on execution he entered 30 and 7 and expected answer was 4 i.e. only integer part not in decimal, but the answer was 4.285 approx., help Raj to correct his expression and achieving the desired output.

>>> He should be doing floor division (//) if he wants the integer part only.

Correct Expression: C = A // B

7. C = -11 % 4; print (C)

>>> The output will be 1

8. Write 2 advantages and disadvantages of Python programming language.

>>> Advantages: 1. Python is a dynamically types language.

2. Python can be integrated with other languages using libraries.

Disadvantages: 1. Python has slow execution speed.

2. Python is relatively a very easy language, which makes programmers commit mistakes such as not adding ‘;’ in other languages which require ‘;’.

9. Identify the valid and invalid identifiers names:

>>> Valid: \_bonus, While, Emp1, Bond007

Invalid: Emp-Code, SrNo., for, #count, 123Go

NOTE: Here, While is a valid identifier name but it can lead to confusions, hence should be avoided

10. Fill in the blanks to execute loop from 10 to 100 and 10 to 1

(i). for i in range(10, 100 + 1):

print(i)

(ii). for i in range(10, 0, -1):

print(i)

11. Which is the valid dictionary declaration?

>>> Valid one is (iii) d3 = {1: 'January', 2: 'February', 3: 'March'}

12. What is the value of x?

>>> x = int(8/3+5)

>>> value of x is: 7

13. What is the output of the following code?

>>> print(“Good”, “Morning”)

>>> “Good Morning”

14. What is the output of the following code?

>>> s = [6, 5, 4, 7, 8, 5]

>>> a.remove(5)

>>> a

Sol: >>> NameError: name 'a' is not defined (line 2)

15. What is the output of the following program?

>>> my\_list = ['p', 'r', 'o', 'b', 'l', 'e', 'm']

>>> print('p' not in my\_list)

>>> False

16. Suppose a tuple T is declared as T = (10, 12, 43, 39). Which of the following is incorrect?

>>> (b) T[2] = -29 [tuple object does not support item assignment]

17. Write a statement in Python to declare a dictionary whose keys are 1, 2, 3 and values are Monday, Tuesday and Wednesday respectively.

>>> dt = {1: “Monday”, 2: “Tuesday”, 3: “Wednesday”}

18. Out of the following find those identifiers, which cannot be used for naming Variable or Functions in a Python Program:

Sol: These cannot be used >>> Days \* Rent, Grand Total, 2Clients, My city

19. Which string method is used to implement the following?

I. To count the number of characters in the string.

>>> len() function (a builtin)

II. To change the first character of the string in capital letter.

>>> .capitalize()

III. To check whether given character is letter or a number.

>>> .isalnum()

IV. To change lowercase to uppercase letter.

>>> .upper()

V. Change one character into another character

>>> .replace()

20. What are the logical operators of Python?

Sol: and, or, not

b Section - B b

1. What is type conversion in Python? What are different types of conversion? Illustrate with example.

>>> Type conversion in Python is the process of conversion of a variable datatype from one datatype to another datatype. Type conversion can either be implicit or explicit. Implicit type conversion is done automatically by the Python interpreter, while explicit type conversion must be done by the coder explicitly.

For example, >>> a = 4

>>> b = 2

>>> c = a / b # Here, type of c gets converted to `float`

And, >>> a = input(“Enter the number to double> ”)

>>> a = float(a)

>>> b = a \* 2 # Here, type of a needed to be converted to `float` explicitly

2. Write a program to enter any number and check it is divisible by 7 or not.

>>> check out sec\_b\_q\_2.py

3. Explain about the tokens in python programming languages

>>> A Token is the smallest individual unit in a python program. All statements and instructions in a program are built with tokens. Tokens can be:

1. Keywords, Python has 33 of them.
2. Identifiers, Variable names we make and use, or
3. Literals. They can be
   1. String Literals – Just any string “njfiuewhfuhwef” enclosed in single, double, or triple (in case of doc string) quotes.
   2. Character Literals – ‘x’, a character enclosed in single or double quotes.
   3. Numeric Literals – A Integer literal, Float literal or Complex literal. Note that python uses letter ‘j’ as the imaginary unit.
   4. Boolean Literals – True and False. They correspond to 1 and 0 Integer Numeric Literal respectively.
   5. Special Literals – None in python has a special meaning. It is not 0, empty sequence or -1. It is used to denote the absence of value.
   6. Literal Collections – List, Tuple, Dictionary or Set
4. Operators. These tokens perform operation in an expression. Operations are applied on *operands*. Operators can be Unary or Binary.
5. Punctuators. Symbols to organize the structure, statements, and expressions. Such as: [ ] { } ( ) @ etc…

4. What is an identifier? What are the rules for declaring the identifiers?

>>> Identifier is a name given to a variable, function, or any other entity in a program. It is used to name and identify them. Python has some restrictions on the naming of identifiers:

1. The name should begin only with lowercased/uppercased letters, or underscore ( \_ ). Digits can only appear other than the starting character. Thus identifiers cannot begin with digits.
2. It can be of any length, only if you like to type loooong names over and over again!
3. It should not be a keyword or reserved word.
4. Name should not use any special characters like !, @, #, $, %, etc. Only underscore ( \_ ) is allowed.

Other than these rules, identifier names should be short, meaningful and descriptive all together.

5. What are the keywords in python programming languages?

>>> Keywords are reserved words in python programming language that can be used by the interpreter only. We cannot use them other than they are intended to be, like for naming things.

6. What are the operators in python programming language?

>>> An operator is used to perform specific mathematical or logical operation on values. The values that the operators work on are called operands. Python allows the use of several operators such as Arithmetic Operators ( + , - , \* , / , // , % , \*\* ), Relational Operators ( == , != , > , < , >= , <= ), Assignment Operators ( = , += , -= , \*= , /= , //= , %= , \*\*= ), Logical Operators ( and , not , or ), Identity operator is and Membership Operator in.

7. What are the features of python programming language?

>>> Features of Python are as follows

1. Python is a high level language.
2. It is a free and open source language.
3. It is an interpreted language, as Python programs are executed by an interpreter.
4. Python programs are easy to understand as they have a clearly defined syntax and relatively simple structure.
5. Python is a beautiful language.
6. Python is case-sensitive. For example, NUMBER and number are not same in Python.
7. Python is portable and platform independent, means it can run on various operating systems and hardware platforms.
8. Python has a rich library of predefined functions.
9. Python is also helpful in web development. Many popular web services and applications are built using Python.
10. Python uses indentation for blocks and nested blocks

8. Write a python program to check the person is eligible or not?

>>> check out sec\_b\_q\_8.py (but beware of the test!)

9. Write a Python Program to Calculate the Area of a Triangle.

>>> check out sec\_b\_q\_9\_a.py and sec\_b\_q\_9\_b.py

10. Python Program to Find the Factorial of a Number.

>>> check out sec\_b\_q\_10.py

11. Explain about the conditional statements in python with flow charts.

>>> Conditional statements in programming languages help us to control the flow of the program based on certain conditions and evaluations. A simple use case would be if we want to greet a user based on its user type or an entirely different authentication system for different types of users in a webapp. Flow charts are created to visualize a proposal of conditional structure before implementation or even of an existing structure to allow other people to understand the essence of the flow control structure.

You can see the examples of flow charts that we created during the development of Chop Chop. We used Flow Charts to help the people new to the project understand how we implemented the Scene Loading System (see: SceneArchitecture\_diagram.jpg) and how the State Machine works (see: StateMachine\_diagram.jpg). They can understand the whole system just by visualizing the diagrams and start contributing right way. The State Machine flow chart also has some legends, where certain shape describe a block with specific meaning.

If we didn’t use Flow Charts, new people may have to dive into hundreds of lines of codes to just understand what’s going on!

12. Write a Python Program to find the largest of three numbers.

>>> check out sec\_b\_q\_12.py

13. Write a python program to reverse the given number using while loop.

>>> check out sec\_b\_q\_13.py

14. Write a python program to Check Prime Number or not.

>>> check out sec\_b\_q\_14.py

15. Write a Python Program to check if a Number is Odd or Even.

>>> check out sec\_b\_q\_15.py

16. Write custom string functions for all existing methods. [\*Important]

>>> check out my\_string\_methods.py

a a a