|  |  |  |  |
| --- | --- | --- | --- |
|  | | | |
| **FT/CSAK/1123/A 12-JUN-2023** | | | |
| **FIRST TERM EXAMINATION - (2023-24)** | | | |
| **SUBJECT: COMPUTER SCIENCE (PYTHON)**  **GRADE: XI (ANSWER KEY)** | | MAX. MARKS: 70TIME: 3 Hrs | |
| **SECTION A** | | | |
| 1. | d) in | | 1 |
| 2. | b) 3 | | 1 |
| 3. | **c) 1111** | | 1 |
| 4. | quit | | 1 |
| 5. | a) 1 | | 1 |
| 6. | a) A>B or A<C | | 1 |
| 7. | b) ROM | | 1 |
| 8. | a) int | | 1 |
| 9. | b)NameError | | 1 |
| 10. | c) List is mutable & Tuple is immutable | | 1 |
| 11. | b) Bit | | 1 |
| 12. | The process of breaking down a big ore complex problem into a set of smaller sub-processes in order to understand a situation better, is known as decomposition | | 1 |
| 13. | c) Utility program | | 1 |
| 14. | **b) 2** | | 1 |
| 15. | **d) end** | | 1 |
| 16. | **c) token** | | 1 |
| 17. | c) dynmic typing | | 1 |
| 18. | d) Statement 2 is True and statement 1 is False1 | |  |
| **SECTION B** | | | |
| 19. | corrected code:  Val = int(input("Value:")) # Error 1  Adder = 0  for C in range(1,Val,3) : # Error 2                Adder+=C                if C%2==0 : # Error 3                        print( C\*10 ) # Error 4                else: # Error 5                      print(C) # Error 6  print (Adder) | | 2 |
| 20. |  | | 2 |
| 21. | import math a = float (input (“Enter first side of triangle :”)) b = float (input (“Enter second side :”)) c = float (input (“Enter third side :”)) s =(a + b + c)/2 area = math.sqrt (s \* (s-a) \* (s-b) \* (s-c)) print (” Area of triangle is “, area)  (1 mark for i/o and 1 mark for calculation) | | 2 |
| 22. |  | | 2 |
| 23. | 122 11 14 9 | | 2 |
| 24. | input units set bill := 0 if units > 250 then calculate bill := units \* 20 elif units <= 100 then calculate bill := units \* 5 else calculate bill := 100 \* 5 + (units – 100) \* 10 end if end if calculate totalbill := bill + 75 print totalbill | | 2 |
| 25. | n=int(input("Enter no:"))  print("Factors of ",n)  for i in range(1,n+1):  if n%i==0:  print(i) | | 2 |
| **SECTION C** | | | |
| 26. | y = int(input("Enter the Year: "))  if y%4==0 and y%100!=0:  print("It is a Leap Year")  elif y%400==0:  print("It is a Leap Year")  else:  print("It is not a Leap Year") | | 3 |
| 27. | for i in range(1,101):  if i%3==0 and i%5==0:  print("FizzBuzz")  elif i%5==0:  print("Buzz")  elif i%3==0:  print("Fizz")  else:  print(i) | | 3 |
| 28. | Draw a flowchart to accept 3 numbers find the smallest of them | | 3 |
| 29. | The break statement enables a program to skip over a part of the code and terminates the very loop it lies within. Execution resumes at the statement immediately following the body of the terminated statement.  Example:  for i in range(1,11):  if i%3==0:  break  else:  print (i)  Output:  1  2  The continue statement continue causes the next iteration by updating the loop with next value in the sequence and the program control passes to the top of the loop.  Example:  for i in range(1,11):  if i%3==0:  continue  else:  print (i)  Output:  1  2  4  5  7  8  10 | | 3 |
| 30. | num=int(input("Enter a number: "))  f=1  for i in range(num,0,-2):  f\*=i  print(num,"!! is: ",f) | | 3 |
| **SECTION D** | | | |
| 31. | 1. Statement 2 2. Statement 6 3. Statement 5 4. Statement 4 5. Statement 1 | | 5 |
| 32. | 1. (84)10 = (1010100)2 2. (2C9)16 = (713)10 3. (101010)2 = (42)10 4. (3674)8 =(11110111100)2 5. (FACE)16= (175316)8 | | 5 |
| 33. | print('''MAIN MENU  -------------------  1. Display Perfect no or not  2. Display Composite no or not  3. Quit''')  ch=int(input("Enter the choice:"))  if ch==1:  n = int(input("Enter any number: "))  sum1 = 0  for i in range(1, n):  if(n % i == 0):  sum1 = sum1 + i  if (sum1 == n):  print("The number is a Perfect number!")  else:  print("The number is not a Perfect number!")  elif ch==2:  n = int(input("Enter any number: "))  for i in range(2, n):  if n % i == 0:  print(n, "is a composite number")  break  else:  print(n, "is not a composite number")  elif ch==3:  exit()  else:  print("Invalid Choice entered") | | 5 |
|  |  | |  |
| 34. | a) [System Software](https://www.geeksforgeeks.org/system-software/) is the type of software that is the interface between application software and the system. Low-level languages are used to write the system software. System Software maintains the system resources and gives the path for application software to run. An important thing is that without system software, the system cannot run. **Application Software** [Application Software](https://www.geeksforgeeks.org/what-is-application-software/) is the type of software that runs as per user request. It runs on the platform which is provided by system software. High-level languages are used to write the application software. It’s a specific purpose software.  **b) Compiler**  Scans the entire program before translating it into machine code.  Syntax errors are found only after the compilation of complete programs.  It takes large amount of time to analyze the source code but overall execution time is faster.  Generates intermediate object code , hence require more memory  **INTERPRETER**  It translates and executes the program line by line.  Syntax errors can be found after translation of every line.  It takes less amount of time to analyze the source code but overall execution time is slower.  No intermediate object code generated, hence require less memory. | | 4 |
| 35. | small=smaller=0  for i in range(10):  n=int(input(“Enter number: “))  if I ==0:  small=n  elif i==1:  if n<=small:  smaller=n  else:  smaller=small  small=n  else:  if n<smaller:  small=smaller  smaller=n  elif n<small:  small=n  print(“The lowest number is :”, smaller)  print(“The second lowest number is : “, small) | | 4 |

\*\*\*