Дискретні структури ІПЗ-21-2

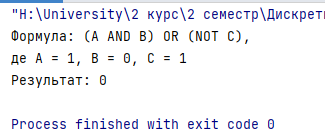
Губарєв Ростислав Вадимович

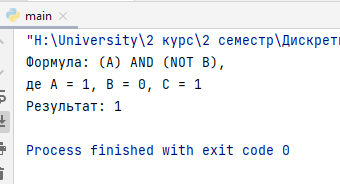
**Лабораторна робота №6.1**

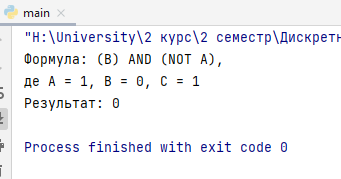
Задано формулу, в якій логічні операції позначені "\*" - AND, "+" - OR, "-" – NOT. Імена змінних починаються з літерних символів. Можуть використовуватися константи «0» та «1». Операції можуть бути згруповані дужками.  
За заданими значеннями змінних, порахуйте формулу та виведіть результат.

|  |
| --- |
| **Код**  def evaluate\_formula(formula, variable\_values):  output\_queue = []  operator\_stack = []   precedence = {'\*': 2, '+': 1, '-': 1}   for char in formula:  if char.isalpha():  output\_queue.append(variable\_values[char])  elif char == '0':  output\_queue.append(0)  elif char == '1':  output\_queue.append(1)  elif char in ['\*', '+', '-']:  while operator\_stack and operator\_stack[-1] != '(' and precedence[char] <= precedence[operator\_stack[-1]]:  output\_queue.append(operator\_stack.pop())  operator\_stack.append(char)  elif char == '(':  operator\_stack.append(char)  elif char == ')':  while operator\_stack and operator\_stack[-1] != '(':  output\_queue.append(operator\_stack.pop())  operator\_stack.pop()   while operator\_stack:  output\_queue.append(operator\_stack.pop())   stack = []  for token in output\_queue:  if token == '-':  operand = stack.pop()  stack.append(int(not operand))  elif token == '\*':  operand2 = stack.pop()  operand1 = stack.pop()  stack.append(int(operand1 and operand2))  elif token == '+':  operand2 = stack.pop()  operand1 = stack.pop()  stack.append(int(operand1 or operand2))  else:  stack.append(token)   return stack.pop()  formula = "(A \* B) + (-C)" variable\_values = {'A': 1, 'B': 0, 'C': 1}  converted = formula.replace("\*", "AND") converted = converted.replace("+", "OR") converted = converted.replace("-", "NOT ")   result = evaluate\_formula(formula, variable\_values) print(f"Формула: {converted},\nде А = 1, В = 0, С = 1") print("Результат:", result) |

**Результат**

****

****

****