

Vārds: _____ Datums: _____

Pārbaudes darbs: Simbolu virķu apstrāde Python valodā

Variants: 1

Darba ilgums: 60 min

Maksimālais punktu skaits: 45

Kāda būs šī koda izvade? Kādas simbolu virķes atšķir programma? Dod vienu piemēru, kad tiks izvadīts "Yes" un kad "No"? (5 punkti)

```
text = "abacbc"
seen = ""
for symbol in text:
    if symbol not in seen:
        seen += symbol

count_ref = text.count(seen[0])
valid = True

for symbol in seen:
    if text.count(symbol) != count_ref:
        valid = False
        break

if valid:
    print("Yes")
else:
    print("No")
```

Atbilde:

Ko dara šis kods? Kas tiks izvadīts? (5 punkti)

```
data = "fly me to the moon"
chunks = data.split()

if len(chunks) > 0:
    measure = len(chunks[len(chunks) - 1])
    print(measure)
else:
    print(0)
```

Atbilde:

Kāds būs šī koda rezultāts? Kādas `text1` un `text2` vērtības iegūs pretēju rezultātu? (5 punkti)

```
text1 = "dog"
text2 = "cat"

if len(text1) != len(text2):
    print(False)
else:
    valid = True

    for symbol in text1:
        if text1.count(symbol) != text2.count(symbol):
            valid = False
            break

    print(valid)
```

Atbilde:

Kas būs mainīgajā t pēc koda izpildes? Kas būtu jāpievieno, lai programma apstrādātu tikai simbolu virknes ar pareizi lietotām iekavām? (5 punkti)

```
text = "abc(de(fg(hi(jk(lm)no)p)qr)st)uv(wx)yz"
stavoklis = 0
t = ""

for char in text:
    if char == "(":
        stavoklis += 1
    elif char == ")":
        stavoklis -= 1
    elif stavoklis % 2 == 0:
        t += char

print(t)
```

Atbilde:

Kāds uzdevums tiek risināts sekojošajā fragmentā? Dod vēl vienu data mainīgā piemēru un rezultātu! (5 punkti)

```
data = "abcfcbaacz"
tracker = ""

for symbol in data:
    if symbol in tracker:
        print(symbol)
        break
    tracker += symbol
```

Atbilde:

Ko izvadīs programma? Kādam jābūt text mainīgajam, lai programma izvadītu -1? (5 punkti)

```
text = "bananagrapesorange"
pattern = "sora"

text_length = len(text)
pattern_length = len(pattern)
result = -1

for pos in range(text_length - pattern_length + 1):
    if text[pos:pos + pattern_length] == pattern:
        result = pos
        break

print(result)
```

Atbilde:

Ko izvadīs programma? (5 punkti)

```
data1 = "abcde"
data2 = "pqr"
result = ""

limit = min(len(data1), len(data2))

for j in range(limit):
    result += data1[j] + data2[j]

result += data1[limit:] + data2[limit:]

print(result)
```

Atbilde:

Ko izvadīs programma? Kādam jābūt data mainīgajam, lai programma izvadītu pretēju rezultātu? (5 punkti)

```
text = "fivebigquackingzephyrsjoltmywaxbed"
pool = "abcdefghijklmnopqrstuvwxyz"
leftover = ""

for symbol in pool:
    if symbol not in text:
        leftover += symbol

if leftover == "":
    print("Yes")
else:
    print("No")
```

Atbilde:

Kāds uzdevums tiek risināts sekojošajā fragmentā? Dod vēl vienu data mainīgā piemēru un rezultātu! (5 punkti)

```
data = "512301000"
pos = len(data) - 1

while pos >= 0 and data[pos] == "0":
    pos -= 1

data = data[:pos + 1]

print(data)
```

Atbilde:
