

AlgoHack : පයිතන් #2



ක්‍රියාකාරක - කොන්දේසි - වක්‍ර ක්‍රියාවලි

කතුවරු
නිරංජන් මීගම්මන, නවින් නමියි

සංස්කරණය
විශ්ව කුමාර



AlgoHack කුඩා අවදියේදී ළමුන්ට පරිගණක විද්‍යාව සහ ක්‍රම ලේඛනය ඉගැන්වීමට ගිලීප සයුර කල ආරම්භයකි. එයට Google for Education සහ ශ්‍රී ලංකා පරිගණක සංගමය සහය ලබා දේ

This work is licensed under a [Creative Commons Attribution-ShareAlike 4.0 International License](https://creativecommons.org/licenses/by-sa/4.0/). Shilpa Sayura Foundation (www.shilpasayura.org)



1.0 ක්‍රියාකාරක (operators)

ගණිත කාර්ම වල යොදාගන්නා $+$, $-$, $*$ හා $/$ සංකේත ගණිත ක්‍රියාකාරකය. ලැයිස්තු සහ කේතවලදී සඳහා ද ක්‍රියාකාරක වේ.

1.1 සාමාජික ක්‍රියාකාරක (Membership operators)

`in` ක්‍රියාකාරක යම් අගයක් ලැයිස්තුවක අඩංගුද යන්න කියයි. පාසලේ නැගිලි ලැයිස්තුවට ඔබ අන්‍යුරුද? යන්න උදාහරණයකි

`x=2`

`y={1,2,3,4,5}` අවයව ලැයිස්තුව

`z=x in y`

`z` සත්‍ය වන්නේ `x` අගය `y` හි සාමාජිකයකු නම් පමණි.

`r= x not in y`

`x` අගය `y` හි සාමාජිකයෙක් නොවේ නම් `r` සත්‍ය වේ

1.2 අනන්‍යතා ක්‍රියාකාරක (Identity operators)

මෙහිදී මනකයේ ස්ථාන දෙකක අන්ති අගයන් සංසන්දනය කෙරේ.

`id(x) == id(y)` ට සමාන නම් `z` සත්‍ය වේ.

`z=x is y`

`id(x) == id(y)` ට අසමාන නම් `r` සත්‍ය වේ.

`r= x is not y`

1.3 ක්‍රියාකාරක ප්‍රමුඛතාව

අගයීම් එකකට වඩා අති ප්‍රකාශණයක ප්‍රථමයෙන් කුමන අගයීම කළ යුතුද යන්න නියම කර ඇත.

ක්‍රියාකාරක පරිපාටියේ ගැටළුව

$3 + 4 * 5$ සලකන්න, පළමුව $3 + 4$ අගයීමෙන් 35 ලැබේ.

$3 + 4 * 5$ සලකන්න, පළමුව $4 * 5$ අගයීමෙන් 23 ලැබේ.

මෙම ගැටළුව විසඳීමට නිවැරදි ස්ථානවල වරහන් යෙදිය යුතුය.

වරහන් යෙදීමේදී වැඩි ප්‍රමුඛතාවයක් අති * හා / මුලින්ම ඇගයේ. අනතුරුව + හා - ඇගයේ. එනම් එකතු කිරීම් හා අඩු කිරීම් කිරීමට පෙර ගුණ කිරීම් හා බෙදීම් කරයි.

ප්‍රතිදානය කුමක්ද?

$$x = 17 / 2 * 3 + 2$$

print (x)

අපි මෙහිදී *, /, +, - පරිපාටියට ඇගයීම සඳහා වරහන් යොදමු.

$$x = (17 / (2 * 3)) + 2$$

පහත ප්‍රකාශන වරහන් යොදා අගයන්න

$$x = 2 + 17 / 2 * 3$$

$$y = 19 \% 4 + 15 / 2 * 3$$

$$z = (15 + 6) - 10 * 4$$

$$p = 17 / 2 \% 2 * 3 ** 3$$

2.0 කොන්දේසි (conditions)

කේතනයක විවිධ තීරණ ගැනීමට කොන්දේසි භාවිතා කෙරේ. කොන්දේසි අනුව ක්‍රියාවලිය වෙනස් මට කරා යොමු කරයි. මේ සඳහා ගණිතමය ප්‍රකාශන සහ සැසඳුම් ප්‍රකාශන භාවිතා කෙරේ.

පහත කොන්දේසි සලකන්න

1. වැස්ස නම්, මට කුඩයක් ඕනෑවේ
2. නිවාඩු දිනයක් නම් මට නව නිදාගත හැකිය,
නැතහොත් උදයෙන් අවදි විය යුතුය.
3. මට මුදල් අඩු නම්, කැමරාවක් මිලදී ගත හැක,
නැතහොත් මිතුරකුගෙන් ඉල්ලා ගත හැක, එසේත්
නැතිනම් මා ඡායාරූප නොගනිමි.

2.1 if කොන්දේසිය

කොන්දේසිය අනුව අදාළ කේත කොටස ක්‍රියාකරවයි.

පයිතන් කේත කොටස් වල ක්‍රියාකාරක ව්‍යුහය පෙන්නීමට කේත **indent** භාවිතා කරයි. කේත **indent** කිරීමට **Tab** යතුර හෝ එහි වෙනුවට **4** ක් භාවිතා වේ. පයිතන් කේත කොටස් වසන නොදමන අතර, කොටස අවසානයේ හිස් පේළියක් තබයි.

```
a=1
```

```
if (a < 3) :  
    print(a)
```

2.2 if ... else කොන්දේසිය

a හා b අගයන් වෙනස් කර බලන්න

a=1

b=2

c=a+b

if (c==3) :

print ('total is 3')

else:

print ('total is not 3')

2.3 බහු කොන්දේසි (Multiple Conditions)

එක කොන්දේසියක් නවත් කොන්දේසියකට යොමු කිරීම සිදුවේ.

If (a > b):

 If (a < d) :

 print(a)

මෙය if ((a > b) and (a<d)) : ලෙසද ලිවිය හැක.

2.4 කුඩු කරන ලද කොන්දේසි (Nested Conditions)

කොන්දේසි ගණනක් මත තීරණ ගැනීමේදී, ප්‍රධාන කොන්දේසියක් තුළ නවත් කොන්දේසියකින් ක්‍රියාවලිය පාලනය කිරීම සිදුවේ.

ඔබ ශ්‍රී ලාංකිකයකු නම්

....ඔබට වයස 18 ට වැඩි නම්

....ඔබ කැමති නම්

....ඡන්දය දැමිය හැක

මෙහිදී ඡන්දය දැමීමට සෑම කොන්දේසියක්ම සත්‍ය විය යුතුය.

if (nationality=="Sri Lankan"):

 if (age > 18):

```
if ( likeToVote==true):  
    canVote=true
```

2.4 ප්‍රතිඵල වර්ගීකරණය සඳහා if ... else ... භාවිතය

```
marks = input("Enter marks: ")  
marks = int(marks)  
if marks >= 75:  
    grade = 'A'  
else:  
    if marks >= 60:  
        grade = 'B'  
    else:  
        grade = 'C'
```

```
print ("Grade is: " + grade)
```

ක්‍රියාවලිය විස්තර කරන්න.

වඩා දියුණු කළ පහත කේතයට 96, 78, 62, 54, 43, 35 සහ 17 අංකයන් කලඥාත් සිදුවන ක්‍රියාවලිය විස්තර කරන්න.

```
marks = input("Enter marks: ")  
marks = int(marks)  
if marks >= 80:  
    grade = 'A'  
else:  
    if marks >= 70:  
        grade = 'B'  
    else:  
        if marks >= 55:
```

```
        grade = 'C'
    else:
        if marks >= 50:
            grade = 'Pass'
        else:
            grade = 'Fail'

print ("Grade is: " + grade)
```

2.5 elif භාවිතය

else if කේන්ද්‍ර ව්‍යුහය සඳහා පමණක් elif පදය භාවිත කරයි.

elif යොදා වෙනස් කළ කේතනය බලන්න.

```
marks = input("Enter marks: ")
```

```
marks = int(marks)
```

```
if marks >= 75 :
```

```
    grade = 'A'
```

```
elif marks >= 60:
```

```
    grade = 'B'
```

```
elif marks >= 50 :
```

```
    grade = 'C'
```

```
elif marks >= 40 :
```

```
    grade = 'S'
```

```
else:
```

```
    grade = 'W'
```

```
print ("Grade is: " + grade)
```

elif යොදාගැනීමේ වාසිය කුමක්ද?

2.6 and හා or සමග කොන්දේසි

ඔබට මොටර් රථයක් තිබීම හා වයස යන සංයුක්ත

කොන්දේසි මත ප්‍රකාශනයක් ප්‍රතිදානය කළ හැක.

↑

```
age = input("Enter age: ")
```

```
age = int(age)
```

```
have_own_car = input("Do you have a car (y/n): ")
```

```
if (age >= 18) and (have_own_car == 'y'):
```



```

print ("You are over 18 years has a car")

if (age >= 18) and (have_own_car == 'n'):
    print ("You are over 18 have no car")

if (age < 18) and (have_own_car == 'y'):
    print ("You are too young to have a car")

if (age < 18) and (have_own_car == 'n'):
    print ("You can buy a car when you are 18")

```

සංයුක්ත බ්ලීස් ප්‍රකාශන 4 ක් ආදාන 2 කින් ප්‍රතිදාන 4 ක් ලබාදේ

if, elif, and , or භාවිතා කර මෙය දියුණු කරන්න.

මේ අයුරින් සැතියේ □□□□ □□ □□□□ □□□□□□
 ආශ්‍රිතව සංයුක්ත පාසල් යාම ගැන ප්‍රකාශන ගොඩ නැගීමට
 ඔබට හැකිද?

3.0 චක්‍ර (Loops)

චක්‍ර මගින් කේත කොටස් නැවත නැවතත් ක්‍රියාත්මක කළ හැකිය.

හිරු නැගීම, ලිදකින් නැම, දිවීම චක්‍ර වේ.

පයිතන් තුළ for හා while චක්‍ර භාවිත වේ.

3.1 while චක්‍රය

while යම් කොන්දේසියක් සත්‍ය වන තුරු ක්‍රියාත්මක වේ.

x = 1

```

while x < 5: # කොන්දේසිය x 5 ට වඩා අඩුවන තුරු
    # පහත කේත 4 වතාවක් ක්‍රියාත්මක වේ

```

```
print (x)
x = x + 1 # x අගය වැඩිවේ
print ('I love python')
```

```
# x අගය 5 ක් වූ විට වක්‍රයෙන් ඉවත්වේ
print ('done')
```

3.2 අගයන් 3 ක සාමාන්‍ය ලබාගැනීමේ වක්‍රය

```
total = 0.0
count = 0 # විචල්‍යයේ මුල් අගය
while count < 3: # කොන්දේසිය
    # අදානය
    number=float(input("Enter a number: "))
    count = count + 1 # විචල්‍යයේ අගය අණක්‍රමණය
    total = total + number # අදාන එකතු කිරීම
```

```
# වක්‍රයෙන් පිටතට පැමිණීම විචල්‍යය > 3 වූ විට
average = total / 3 # එකතුව බෙදීම
# ප්‍රතිදානය
print ("The average is " + str(average))
```

3.3 For වක්‍රය

විධාන කරන නිශ්චිත වාර ගණනක් ක්‍රියාකරයි.
පහත කේතනයේ ක්‍රියාකරන වාර ගණන 10 කි.

```
for i in range (10):
    print (i)
```

පරිතන් අණක්‍රමයනය 0 සිට පටන් ගනී.
10 අගය ලබාදුන් විට 0 සිට 9 දක්වා වාරයක් ක්‍රියාත්මක වේ.

චක්‍රය සම්පූර්ණ වනවිට i අගය ස්ථයංක්‍රීයව අනුක්‍රමණය වේ.

```
# (5,10) අගය ලබාදුන් විට 5 සිට 9 දක්වා ක්‍රියාත්මක වේ
for c in range (5,10):
    print (c)
```

```
අගයන්වල එකතුව ගණනය කිරීම
sum=0
for i in range (10):
    sum=sum+ i # අගය
print(sum)
```

3.4 අතරමැද කැඩීම (Break) භාවිතය

චක්‍රයක ක්‍රියාවලිය අතරමැද නවතා චක්‍රයෙන් පිටතට යොමුකිරීමට භාවිතා කරයි. ස්ථානය කරන විට වතුර නැවතීමට සමාන වේ.

3.5 for චක්‍රය අතරමැද කැඩීම

```
for c in range (1,6):
    if c == 3:
        break # මේ චක්‍රයෙන් එලියට යන්න
    print (c)
```

3.6 නොනවතින while චක්‍රය කැඩීම

```
while 1:
    print ('Running? ')
    print ('Press the Ctrl + C')
    print ('to stop running')
```

```
print ('I am out of loop')
print ('done')
```

3.7 අදානය මත වක්‍රයක් වක්‍රය කැඩීම

while 1:

```
    print ('running')
```

```
    answer = input('Press y to end')
```

```
    if answer == 'y':
```

```
        print ('you stopped me?')
```

```
        break # වක්‍රයෙන් ඉවත් වේ
```

```
print ('Out of loop')
```

3.8 ඊ ළඟ වාරයට (Continue) යොමු කිරීම

ක්‍රියාත්මක වක්‍රයේ අතර මැද නවතා ඊ ළඟ වාරයට යොමු කරයි. වක්‍රයෙන් ඉවත් වීමක් නැත. ක්‍රියාවලි ද්‍රව්‍යදී දැවී යාමේදී ඊ ළඟ පිතිකරුපැමිණීමට සමාන වේ.

3.9 for සමග continue

```
for letter in 'Python':
```

```
    if letter == 'h':
```

```
        # h අකුර ප්‍රතිදානය නොවේ. ඊ ළඟ වාරයට යන්න
```

```
        continue
```

```
    print 'Current Letter :', letter
```

3.10 while සමග continue

```
var = 10
```

```
while var > 0:
```

```
    var = var -1
```

```
    if var == 5:
```

```
        continue # අගය 5 වට ඊ ළඟ වාරයේ මුලට යන්න
```

```
    print 'Current variable value :', var
```

```
print "Loop Ends!"
```

අනහරන්න (Pass)

Pass ක්‍රමලේඛ කේෂි අනහර යාමට භාවිතා කරයි

```
for letter in 'Python':
    if letter == 'h':
        pass # අනහර යන්න
    print ('This is pass block')
    print ('Current Letter :', letter)
print "End!"
```

3.11 වක්‍ර තුළ වක්‍ර

බයිසිකලයක් පැදීමේදී වක්‍ර ගණනාවක් ක්‍රියාත්මකය.

පැඩල වක්‍රය විසින් ප්‍රධාන දැනී රෝදය එක වාරයක් කරකවන විට, කුඩා දැනී රෝදය වාර ගණනක් කැරකේ.

මෙයින් ලැබෙන ප්‍රතිදානය කුමක්ද?

```
for i in range(10): # වාර 10 ක් කරන්න
    for j in range(10): # වාර 10 ක් කරන්න
        print ((i * 10) + j)
```

සෑම i වාරයකටම j වාර 10 ක් ක්‍රියාත්මකය.

සමස්ත ප්‍රතිදාන වාර 100 ක් වේ. කේතය විස්තර කරන්න.

3.12 while සඳහා වක්‍ර තුළ වක්‍ර

```

x = 1
while (x < 6): # පිටත වක්‍රය
    print ("\n") # හිස් පෙළියක් ඇති කිරීම
    print ("x = " + str(x))
    x = x + 1
    y = 1
    while (y < 6): # ඇතුළත වක්‍රය
        print ("y = " + str(y))
        y=y+1
        z= x * y
    print (x)
    print(y,z)

```

ඉහත කේන් ක්‍රියාව ඔබට විස්තර කළ හැකිද?

3.14 range() ක්‍රියාකාරකය

range([start], stop[, step])
 range() ශ්‍රිතය සඳහා පරාමිති 3 කි.
 සියලු පරාමිති පූර්ණ සංඛ්‍යා වේ.
 start: ආරම්භක වාරයේ අගය
 stop: අවසාන වාරයට පසු අගය
 step: වාර අනුක්‍රමණ පරතරය
 පළමු හා අවසාන පරාමිති අවශ්‍යම නොවේ.
 පරාමිතිය සෘණ අගයක් වූ විට වාර සංඛ්‍යා රේඛාවක ආකාරව
 ක්‍රියාත්මක වේ..

range(0, 5) 0 සිට 5 දක්වා, 5 ඇතුළත් නොවේ

පහත ප්‍රතිදානයන් කුමක් විය හැකිද?

```
for i in range(5):  
    print(i)
```

```
for i in range(3,6):  
    print(i)
```

```
for i in range(2,10,2):  
    print(i)  
for i in range(-2,6,2):  
    print(i)
```

3.15 ප්‍රථමක සංඛ්‍යා ප්‍රතිදානය

ප්‍රතිදාන විස්තර කරන්න?

```
for num in range(10,20): #10 න් 20 න් අතර ඇති සංඛ්‍යා  
    for i in range(2,num): # සංඛ්‍යාවේ factors  
        if num%i == 0: # num අගය i මගින් ඉතිරි නැතිව  
            බෙදේනම්  
                j=num/i # දැන් බෙදෙන්න  
                print '%d equals %d * %d' % (num,i,j)  
                break # i වක්‍රයෙන් මිදෙන්න  
    else:  
        print num, 'is a prime number'
```



කැමරා චිත්‍රය
කැමරා වෙලාවක
කැමරා තැනක
නිදහසේ ඉගෙන ගන්න
පාඩම් සහ ප්‍රශ්න
Shilpa64.lk



AlgoHack කුඩා අවදියේදී ළමුන්ට පරිගණක විද්‍යාව සහ ක්‍රම ලේඛනය ඉගැන්වීමට ශිල්ප සයුර කල අරමුණක්. එයට Google for Education සහ ශ්‍රී ලංකා පරිගණක සංගමය සහය ලබා දේ

AlgoHack : පයිතන්#2



ක්‍රියාකාරක - කොන්දේසි - චක්‍ර ක්‍රියාවලි