

# மேல்நிலை இரண்டாம் ஆண்டு கணினி அறிவியல்

## ASSIGNMENT – 1

சிக்கலைத் தீர்க்கும் உத்திகள்



**ஜெ. கவிதா** B.Sc, B.Ed, M.C.A, M.Phil.,

கணினி பயிற்றுநர் நிலை - I

அரசு மேல்நிலைப்பள்ளி, சர்க்கார்சாமக்குளம்,

கோயம்புத்தூர் - 641107.

## சிக்கலைத் தீர்க்கும் உத்திகள்

### செயற்கூறுகள்(Functions) - அறிமுகம்

- நிரலாக்கமொழியின் கூற்றுகளைப் பயன்படுத்தி நெறிமுறைகள் செயல்படுத்தப்படுகின்றன.
- ஒன்றுக்கு மேற்பட்ட கூற்றுகள், பலமுறை மீண்டும் மீண்டும் செய்யப்பட வேண்டும் எனில், அந்த செயலைச் செய்து முடிப்பதற்காக துணைநிரல்கள் (Subroutines) பயன்படுகின்றன. நிரலாக்கமொழிகளில் இத்துணைநிரல்கள் செயற்கூறுகள் (Functions) என்று அழைக்கப்படுகின்றன.

### அளபுருக்கள் மற்றும் செயலுருபுக்கள்

- அளபுருக்கள் என்பது செயற்கூறு வரையறையில் உள்ள மாறிகள் ஆகும்.
- செயற்கூறு வரையறைக்கு அனுப்பப்படும் மதிப்புகள் செயலுருபுக்கள் ஆகும்.

### தரவு வகை இல்லா அளபுருக்கள்

- செயற்கூற்றில் மாறிகளின் தரவுவகை குறிப்பிடப்படவில்லையெனில் நிரல் பெயர்ப்பிகள் தரவு வகையை நிரல் நெறிமுறைப்படி சரி செய்கிறது.

```
எ.கா. let rec pow a b:=
    if b=0 then 1
    else a * pow b (a-1)
```

இங்கு b என்பது அளபுருவாகும். b -ன் தரவினம் குறிப்பிடவில்லையென்றாலும் செயற்கூறு வரையறையில் உள்ள then கிளை 1 என்ற மதிப்பை திருப்பி அனுப்புவதால் நிரல் நெறிமுறைப்படி கோவை முழுவதும் int தரவு வகையாகும். மேலும் a -ன் மதிப்பு int தரவு வகையான b யுடன் பெருக்கப்படுவதால் a -ன் தரவு வகையும் int ஆகும்.

### தரவு வகையுடன் கூடிய அளபுருக்கள்:

- தரவு வகை வெளிப்படையாக குறிப்பிடப்படும்போது தரவு வகைகளை சார்ந்த தருக்கப்பிழைகள் தவிர்க்கப்படுகிறது.

```
எ.கா. let rec pow (a: int) (b: int) : int :=
    if b=0 then 1
    else a * pow b (a-1)
```

a மற்றும் b - ன் தரவு வகையினைக் குறிப்பிடும் போது அடைப்புக்குறிக்குள் ( ) குறிப்பது அவசியமாகிறது.

### இடைமுகம் மற்றும் செயல்படுத்துதல் வேறுபாடு:

இடைமுகம்	செயல்படுத்துதல்
ஒரு பொருள் செய்யக்கூடிய நடவடிக்கையை வரையறுக்கிறது. ஆனால் அவற்றை உண்மையில் செய்யக்கூடியதில்லை.	இடைமுகத்தில் வரையறுக்கப்பட்டுள்ள கட்டளைகளை நிறைவேற்றுகிறது.

### இடைமுகத்தின் பண்புகூறுகள்:

- ஒரு பொருளை முறையாக உருவாக்கி வழங்குவதற்கும் அதனை செயல்படுத்துவதற்கும் தேவையான இடைமுகத்தை இனக்குழு வார்ப்புரு குறிப்பிடுகிறது.
- செயற்கூறுகளைப் பொருளுக்கு அனுப்புவதன் மூலம் பொருளின் பண்புகளையும், பண்புகூறுகளையும் கட்டுப்படுத்த முடிகிறது.

### Pure செயற்கூறுகள்:

- ஒரே மாதிரியான அளபுருக்களை அனுப்பும் போது, சரியான விடையைத் தரும் செயற்கூறு pure செயற்கூறுகள் ஆகும்.
- மாறியின் பண்பை மாற்றக்கூடிய எந்த விதமான வெளிப்புற மாறியும் இல்லாமல் இருந்தால் அந்த செயற்கூறு pure செயற்கூறாகும்.

எ.கா : let square x

x \* x

மேலேயுள்ள square செயற்கூறு Pure செயற்கூறு ஆகும். ஏனென்றால் ஒரே மாதிரியான உள்ளீட்டிற்கு வேறு வித்தியாசமான வெளியீட்டை தராது.

### Impure செயற்கூறுகள்:

- செயற்கூறுக்கு அளபுருக்களை அனுப்பாத போதும், செயற்கூறின் உள்ளே உள்ள மாறியானது பக்கவிளைவுகளை ஏற்படுத்தும். இந்த வகையான செயற்கூறை Impure செயற்கூறு என்கிறோம்.
- ஒரு செயற்கூறு அந்த வரையறைத் தொகுதியின் வெளியே உள்ள மாறியை ஒவ்வொரு முறை அழைக்கும் பொழுதும் செயற்கூறு ஒரே மாதிரியாக இயக்கப்படும் என்று கூற இயலாது.

எ.கா: let Random number

let a := random()

if a > 10 then return: a

else return: 10

இதில் random என்பது impure செயற்கூறு ஆகும். ஏனெனில் இதனை அழைக்கும்போது என்ன விடை கிடைக்கும் என நிச்சயமாக கூற முடியாது.

### தரவு அருவமாக்கம்

- அருவமாக்க தரவு வகை என்பது பொருள்களுக்கான வகை அல்லது இனக்குழு ஆகும். செயல்படுத்தப்படும் விவரங்களை மறைத்து அவசியமானவற்றை மட்டும் வழங்கும் செயல்முறையை அருவமாக்கம் என்கிறோம்.
- தரவு அருவமாக்கிகளை செயல்படுத்த, ஆக்கிகள் மற்றும் செலக்டர்கள் என்ற இரண்டு செயற்கூறுகள் உருவாக்கப்பட வேண்டும்.
  - **ஆக்கி செயற்கூறுகள்** அருவமாக்கம் தரவு வகையை கட்டமைக்கப் பயன்படுகிறது.
  - **செலக்டர் செயற்கூறுகள்** தகவல்களை தரவு வகையிலிருந்து பெறுவதற்கு பயன்படுகிறது.

### List

- List என்பது, கோவைகளை சதுர அடைப்புக்குறிக்குள் காற்புள்ளியில் பிரிக்கப்பட்டிருக்கும் அமைப்பு ஆகும். List பல மதிப்புகளை சேமிக்கும். இம்மதிப்புகள் எவ்வகையாகவும் இருக்கலாம் அல்லது மற்றொரு List ஆகவும் இருக்கலாம். List - ல் இரு மதிப்புகளை ஒன்றாக இணைக்கும் செயல்முறையை pairs என்று அழைக்கிறோம். List அதிகமாக பயன்படுத்தும் முறையாகும்
- **எ.கா:** lst[(0, 10), (1, 20)] இதில், lst[0] = 10, lst[1] = 20
- இவ்வாறு இட மதிப்பும், உறுப்பின் மதிப்பும் ஒன்றாக இணைக்கப்படுவதால் List, pairs என்று அழைக்கப்படுகிறது.

### Tuples

- Tuples என்பது அடைப்புக்குறிக்குள் கொடுக்கப்பட்டுள்ள தொடர் மதிப்புகளை காற்புள்ளியில் பிரிக்கப்பட்டிருக்கும் அமைப்பு ஆகும்.
- **எ.கா:** colour= ('red', 'blue', 'Green')

## வரையெல்லை

- வரையெல்லை என்பது மாறிகள், அளபுருக்கள் மற்றும் செயற்கூறுகளின் அணுகியல்பை நிரலின் ஒரு பகுதியில் இருந்து மற்றொரு பகுதிக்கு குறிப்பதாகும். அதாவது, நிரலின் எந்தப் பகுதியை அணுக அல்லது பயன்படுத்த முடியும் என்பதைக் குறிப்பதாகும்.

## மாறியின் வரையெல்லைகளின் வகைகள்

- வரையெல்லை என்பது சரியான மதிப்பை பெறுவதற்காக மாறிகள் எந்த வரிசையில் பொருளுடன் Map செய்யப்பட வேண்டும் என்பதை வரையறுக்கிறது. LEGB விதி வரையெல்லை தேடப்பட வேண்டிய வரிசையை தீர்மானிக்கப் பயன்படுகிறது.
- **உள்ளமை வரையெல்லை (Local Scope):** உள்ளமை வரையெல்லை என்பது நடப்பு செயற்கூறில் வரையறுக்கப்பட்ட மாறிகளைக் குறிக்கும். செயற்கூறு எப்பொழுதும் மாறியின் பெயரை முதலில் அதன் உள்ளமை வரையெல்லையில் பார்வையிடும். அந்த வரையெல்லையில் இல்லையென்றால் மட்டுமே வெளி வரையெல்லையில் சோதிக்கும்.
- **அடைக்கப்பட்ட வரையெல்லை (Enclosed):** மற்றொரு செயற்கூறின் வரையறையை, தன்னுள் கொண்ட ஒரு வெளி செயற்கூறினுள் ஒரு மாறி அறிவிக்கப்பட்டால், உள் செயற்கூறானது, வெளி செயற்கூறினுள் உள்ள மாறிகளை அணுக முடியும். இதுவே அடைக்கப்பட்ட வரையெல்லை எனப்படும்.
- **முழுதளாவிய வரையெல்லை (Global):** நிரலின் அனைத்து செயற்கூறுகளுக்கும் வெளியே அறிவிக்கப்பட்ட மாறிகள் முழுதளாவிய மாறிகள் எனப்படும். முழுதளாவிய மாறிகளை நிரலின் அனைத்துச் செயற்கூறுகளும் உட்புறமும், வெளிப்புறமும் அணுக முடியும்.
- **உள்ளிணைந்த வரையெல்லை (Built - in):** நிரலாக்க மொழியின் நூலக செயற்கூறினுள் வரையறுக்கப்பட்ட மாறி உள்ளிணைந்த வரையெல்லையைக் கொண்டிருக்கும். இவைகள், நூலக கோப்புகள் நிரலில் செயல்பட தொடங்கியவுடன் இயக்கப்படும்.

எ.கா: x = 10

```
z = 30 #Global
def outer():
    y = 20 #Enclosed
    def inner():
        x = 40 #Local
        print(f'x is {x}')
        print(f'y is {y}')
        print(f'z is {z}')
        print(len("abc")) #Built-in
    inner()
outer()
#Output:
x is 40
y is 20
z is 30
3
```

**வெளியீடு:**

## நெறிமுறையின் யுக்திகள்

- நெறிமுறை என்பது குறிப்பிட்ட செயலை நிறைவேற்றுவதற்காக வரையறுக்கப்பட்ட கட்டளைகளின் தொகுப்பாகும்.
- கொடுக்கப்பட்ட சிக்கலைத் தீர்க்கும் படிநிலை உடைய செய்முறை ஆகும்.
- நெறிமுறையை எந்தவொரு பொருத்தமான நிரலாக்க மொழியிலும் செயல்படுத்த முடியும்.

## நெறிமுறையின் சில முக்கிய செயல்கள்

- தேடல்: தரவு கட்டமைப்பில் உள்ள ஒரு உருப்படியைத் தேடும் செயல்முறை தேடல் எனப்படும்.  
அதன் வகைகள்: 1. வரிசைமுறைத்தேடல் 2. இருமத்தேடல்
- வரிசையாக்கம்: ஒரு குறிப்பிட்ட வழிமுறைகளைப் பயன்படுத்தி தரவுகளை ஏறுவரிசையிலோ அல்லது இறங்கு வரிசையிலோ வரிசைப்படுத்துவது வரிசையாக்கம் எனப்படும்.  
அதன் வகைகள்: 1. குமிழி வரிசையாக்கம் 2. தேர்ந்தெடுப்பு வரிசையாக்கம் 3. செருகும் வரிசையாக்கம்.
- செருகுவது: தரவு கட்டமைப்பில் ஒரு உருப்படியை செருகுவது
- மேம்படுத்துதல்: தரவு கட்டமைப்பில் ஒரு உருப்படியை மேம்படுத்துவதல்
- நீக்குதல்: தரவு கட்டமைப்பில் ஒரு உருப்படியை நீக்குதல்

## நெறிமுறையின் பண்பியல்புகள்

உள்ளீடு	பூஜ்யம் அல்லது அதிக எண்ணிக்கையில் வழங்கப்படவேண்டும்
வெளியீடு	குறைந்த பட்சம் ஒன்றாவது உருவாக்கப்பட வேண்டும்
எல்லையற்றது	குறிப்பிட்ட எண்ணிக்கையிலான படிநிலைகளில் நெறிமுறை நிறுத்தப்பட வேண்டும்
வரையறுத்தல்	அனைத்துச் செயல்பாடுகளும் நன்றாக வரையறுக்கப்பட வேண்டும்
செயல்தன்மை	ஒவ்வொரு கட்டளைகளும் திறம்பட செயல்படுத்தப்பட வேண்டும்
உண்மைத்தன்மை	நெறிமுறைகள் பிழை இல்லாததாக இருக்க வேண்டும்
எளிமை	செயல்படுத்த மிக எளிதாக இருக்க வேண்டும்
குழப்பமற்றது	நெறிமுறையானது தெளிவாகவும், குழப்பமற்றதாகவும் இருக்க வேண்டும்
செயலாக்கம்	கிடைக்கும் வளங்களை வைத்து செயல்பட வேண்டும்
அடக்கமானது	பொதுவானதாக இருக்க வேண்டும்
சார்பற்றது	நெறிமுறையானது எந்த நிரலாக்க மொழியையும், இயக்க அமைப்பையும் சாராமல் இருக்க வேண்டும்.

## நெறிமுறையின் சிக்கல்

- கணினி வளங்கள் வரம்புக்குட்பட்டது. நேரம் மற்றும் இடச்சிக்கலைப் பயன்படுத்துவதன் மூலம் நெறிமுறையின் செயல்திறன் வரையறுக்கப்படுகிறது.
  - நேரச்சிக்கல்: நெறிமுறை ஒரு செயலைச் செய்து முடிக்க தேவைப்படும் கட்டளைகளின் எண்ணிக்கையே நெறிமுறையின் நேரச்சிக்கல் எனப்படும்.
  - இடச்சிக்கல்: ஒரு நெறிமுறையின் செயல்பாடு முடியும் வரை அதற்கு தேவைப்படும் நினைவக இடமே இடச்சிக்கல் எனப்படும்.

## Asymptotic குறியீடுகள்

- Asymptotic குறியீடுகள் நேரம் மற்றும் இடச்சிக்கலைப் பற்றிய அர்த்தமுள்ள கூற்றுகளைப் பயன்படுத்தும் ஒரு மொழியாகும்.
- மூன்று Asymptotic குறியீடுகள் நெறிமுறையில் நேரச்சிக்கலை குறிக்க மிகவும் பயன்படுகிறது. அவை, 1. Big O 2. Big Ω 3. Big Θ





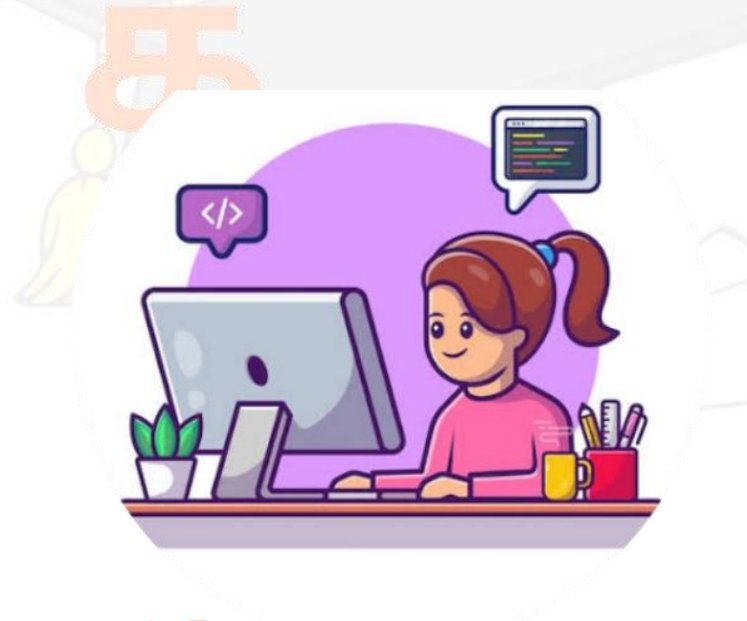
**கண் போன்ற கல்வியை**



**நீ பொன் போல பாதுகாத்தால்**

**மண்ணுலகில்**

**சான்றோனாய் வாழ்ந்து  
விண்ணைத் தொடலாம்...!**



**J. Kavitha B.SC.,B.Ed.,M.C.A.,M.Phil**

**Computer Instructor Gr-1**

**GHSS, Sarkarsamakulam**

**Coimbatore - 641107.**