

# பிரச்சினைகளைத் தீர்ப்பதற்குச் செய்நிரல்களை எழுதுதல்

இப் பாடத்தைக் கற்ற பின்னர் நீங்கள்,

- ஒரு பிரச்சினையைப் பகுப்பாய்வு செய்தலும் ஒரு பிரச்சினைக்கான நெறிமுறையை உருவாக்கலும்
  - நிருவாகக் கட்டமைப்பும் அதன் தேவையும்
  - பாய்ச்சற் கோட்டுப்படங்களை வரைதலும் போலிக் குறிமுறையை எழுதுதலும் அவற்றுக்கிடையே மாற்றலும்
  - ஒரு பிரச்சினைக்கு மாற்றுத் தீர்வுகளைக் காணல்
  - மாறிகளும் மாறிலிகளும்
  - பஸ்கால் செய்நிரல் மொழி
  - செய்நிரல் மொழிகளின் பரிணாமம்
- என்பன பற்றிய விளக்கத்தைப் பெறுவீர்கள்.

## 1.1 பிரச்சினையைப் பகுப்பாய்வு செய்தல்

ஒரு பிரச்சினையைத் தீர்த்துப் பெறப்படும் பேறு வெளியீடு (Output) எனப்படும். அவ்வெளியீட்டைப் பெறுவதற்கு வழங்கப்படும் தரவுகள் உள்ளீடு (Input) எனப்படும். உள்ளீட்டை வெளியீடாக மாற்றும் பொறிமுறை முறைவழியாக்கம் (Processing) எனப்படும். முறைவழியாக்கம் படிமுறைகளாக நடைபெறும் அதேவேளை அவற்றை ஒழுங்குமுறையாகக் காட்டல் மிகவும் முக்கியமானது. ஒரு பிரச்சினையைப் பகுப்பாய்வு செய்யும்போது அதில் உள்ளீடு, முறைவழியாக்கம், வெளியீடு ஆகியன வேறுவேறாக இனங்காணப்படும்.

**உதாரணங்கள்**

**பிரச்சினை 1 :** அஞ்சலில் இடுவதற்கு உகந்த ஒரு கடிதத்தைத் தயாரித்தல்

உள்ளீடு : கடிதத்தை எழுதுவதற்கு உகந்த ஒரு தாளும்  
பேனாவும்  
ஒரு கடிதவுறையும் முத்திரையும் பிசினும்



- முறைவழியாக்கம் :
1. கடிதத்தை எழுதுதல்
  2. கடிதத்தை மடித்துக் கடிதவுறையில் இடுதல்
  3. கடிதவுறையை ஒட்டுதல்
  4. கடிதவுறையில் முகவரியை எழுதுதல்
  5. முத்திரையை ஒட்டுதல்

வெளியீடு : அஞ்சலில் இடுவதற்கு உகந்த நிலைமையில் இருக்கும் ஒரு கடிதம்

குறிப்பு : இம்முறைவழியில் படிமுறை 4 ஐயும் படிமுறை 5 ஐயும் மாற்றலாம். ஆனால் ஏனைய படிமுறைகளை ஒழுங்கு முறையாகச் செய்தல் வேண்டும்.

**பிரச்சினை 2 :** சுவையான தேநீர் தயாரித்தல்

உள்ளீடு : தேயிலைத் தூள், சீனி, வெந்நீர்

முறைவழியாக்கம் :

1. தேயிலைத் தூளை வடியில் இடுதல்
2. வடியினூடாக வெந்நீரைக் கிண்ணத்தில் ஊற்றுதல்
3. ஒரு தேக்கரண்டியளவு சீனியைக் கிண்ணத்தில் இடுதல்
4. கரண்டியினால் தேநீரைக் கலத்தல்
5. தேநீரில் சிறிதளவை எடுத்துச் சுவையைச் சோதித்தல்
6. சுவை போதுமானதாக இராவிட்டால் படிமுறைகள் 3, 4, 5 ஆகியவற்றை மறுபடியும் செய்தல்



வெளியீடு : சுவையான தேநீர்

**பிரச்சினை 3 :** புத்தகப் பொதி ஒன்றில் 40 பக்க, 80 பக்கப் புத்தகங்களை ரேவதி, கீதா என்ற இரு சகோதரிகளுக்கு முறையே விநியோகித்தல்

உள்ளீடு : புத்தகப் பொதி

முறைவழியாக்கம் :

1. புத்தகப் பொதியைத் திறத்தல்
2. பொதியிலிருந்து ஒரு புத்தகத்தை எடுத்தல்
3. பக்க எண்ணிக்கை 40 எனின் ரேவதிக்குக் கொடுத்தல்
4. பக்க எண்ணிக்கை 80 எனின் கீதாவிற்குக் கொடுத்தல்
5. பொதியில் உள்ள புத்தகங்கள் முடியும் வரைக்கும் படிமுறைகள் 2, 3, 4, 5 ஐச் செய்தல்



வெளியீடு : ரேவதிக்கு 40 பக்கப் புத்தகங்கள் கிடைத்தல், கீதாவுக்கு 80 பக்கப் புத்தகங்கள் கிடைத்தல்

**பிரச்சினை 4 :**

இரு எண்களின் கூட்டுத்தொகையைக் காணல்

உள்ளீடு : இரு எண்கள்

முறைவழியாக்கம் : இரு எண்களைக் கூட்டல்

வெளியீடு : கூட்டுத்தொகை

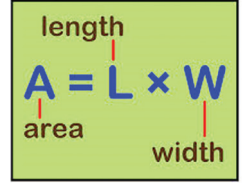
**பிரச்சினை 5 :**

ஒரு செவ்வகத்தின் பரப்பளவைக் காணல்

உள்ளீடு : செவ்வகத்தின் நீளமும் அகலமும்

முறைவழியாக்கம் : பரப்பளவு = நீளம் × அகலம்

வெளியீடு : பரப்பளவு

**பிரச்சினை 6 :**

இரு எண்களில் பெரிய எண்ணைக் காணல்

உள்ளீடு : இரு எண்கள்

முறைவழியாக்கம் : இரு எண்களையும் ஒப்பிடுவதன் மூலம் பெரிய எண்ணைக் கண்டறிந்து வெளிப்படுத்தல்

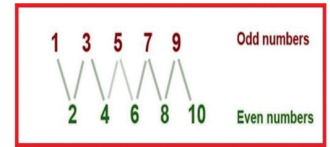
வெளியீடு : பெரிய எண்

**பிரச்சினை 7 :**

ஒர் எண் ஒற்றையா , இரட்டையா எனக் காணல்

உள்ளீடு : எண்

முறைவழியாக்கம் : எண்ணை இரண்டினால் வகுத்து மீதியைக் காணல்



மீதி = 0 எனின் இரட்டை எண் எனத் தீர்மானித்தல்

மீதி = 1 எனின் ஒற்றை எண் எனத் தீர்மானித்தல்

வெளியீடு : எண்ணின் வகை (ஒற்றை அல்லது இரட்டை எனக் காணல்)

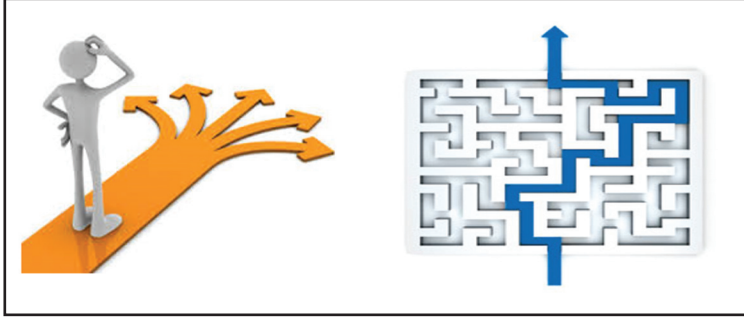
## செயற்பாடு



1. 100 இனிப்புகளை 20 பேரிடையே சமமாகப் பிரித்தல் தொடர்பான உள்ளீடு, முறைவழியாக்கம், வெளியீடு ஆகியவற்றைக் காட்டுக.
2. ஒரு பட்டத்தைச் செய்வதைக் காட்டுவதற்கு உள்ளீடு, முறைவழியாக்கம், வெளியீடு ஆகியவற்றைக் காட்டுக.

### மாற்றுத் தீர்வுகளை இனங்காணல்

ஒரு பிரச்சினைக்கு ஒன்றுக்கு மேற்பட்ட தீர்வுகள் இருந்தால் இவை மாற்றுத் தீர்வுகள் எனப்படும். இது பிரச்சினையின் இயல்பினைப் பொறுத்தது.



உரு 1.1

### உதாரணம்

நீங்கள் ஒரு பாடசாலைப் பேருந்தில் பாடசாலைக்கு வருவதாகக் கொள்க. பாடசாலைக்கு வரும்போது பேருந்தில் உள்ள ஒரு குறைபாடு காரணமாக பேருந்து இடையில் நின்றால் நீங்கள் பாடசாலைக்கு வரத்தக்க வேறு வழிமுறைகள் பற்றிச் சிந்தித்துப் பார்ப்பீர்கள் அல்லவா? இவ்வாறு நீங்கள் பாடசாலைக்குச் செல்லத்தக்க வேறு மாற்றுத் தீர்வுகள் பற்றிச் சிந்திப்போம்.

1. பாடசாலைக்கு வரும் வேறொரு பாடசாலைப் பேருந்தில் பாடசாலைக்கு வருதல்
2. உங்களிடம் பணம் இருந்தால் ஓர் இ. போ. ச அல்லது தனியார் பேருந்தில் பாடசாலைக்கு வருதல்
3. வீதி வழியே நடந்து சென்று பாடசாலைக்கு வருதல்
4. ஒரு குறுகிய வழியில் நடந்து சென்று பாடசாலைக்கு வருதல்
5. பெற்றோருக்கு இதனை அறிவிக்கமுடியுமாயின் அவர்களுடைய உதவியுடன் பாடசாலைக்கு வருதல்
6. நம்பிக்கையான ஒருவரின் உதவியுடன் வாகனத்தில் அல்லது மோட்டர்ச் சைக்கிளில் பாடசாலைக்கு வருதல்

அந்நாள் கட்டாயம் பாடசாலைக்கு வரவேண்டிய ஒருநாள் எனின் இம்மாற்றுத் தீர்வுகளிடையே நல்ல ஒரு தீர்வைத் தெரிந்தெடுப்பீர்கள் அல்லவா?

அதற்கேற்ப ஒரு பிரச்சினைக்குத் தீர்வுத் தொடை இருக்குமெனின், அதனைக் கருத்திற் கொண்டு ஒரு நல்ல தீர்வைத் தெரிந்தெடுத்தல் மிகவும் உகந்தது.

இவ்வாறு ஒரு பிரச்சினைக்கு இருக்கும் எல்லாத் தீர்வுகளும் தீர்வு வெளி (Solution Space) எனப்படும். கணினிச் செய்நிரலிலும் பல்வேறு தீர்வுகளை இனங்கண்டு ஒரு நல்ல தீர்வைத் தெரிந்தெடுத்தல் வேண்டும். அப்போது ஓர் எளிய குறுகிய செய்நிரலை அமைக்க முடியும்.

### உதாரணம் 1

ஒரு செவ்வகத்தின் சுற்றளவைக் காண்பதற்கு உள்ள தீர்வு வெளியைக் காண்போம்.

இப்பிரச்சினைக்குரிய உள்ளீடு, வெளியீடு, முறைவழியாக்கம் ஆகியவற்றைப் பகுப்பாய்வு செய்து காட்டுவோம்.

உள்ளீடு : செவ்வகத்தின் நீளமும் அகலமும்

முறைவழியாக்கம் : சுற்றளவைக் கணித்தல்

வெளியீடு : சுற்றளவைக் காட்டல்

சுற்றளவைக் கணிப்பதற்கான தீர்வு வெளியைக் காண்போம்.

தீர்வு 1 : சுற்றளவு = நீளம் + அகலம் + நீளம் + அகலம்

தீர்வு 2 : சுற்றளவு =  $(2 \times \text{நீளம்}) + (2 \times \text{அகலம்})$

தீர்வு 3 : சுற்றளவு =  $2 \times (\text{நீளம்} + \text{அகலம்})$

இத்தீர்வுகளிடையே கூட்டல் பற்றிய விளக்கம் மாத்திரம் உள்ள ஒருவர் நல்ல தீர்வாகத் தீர்வு 1 ஐத் தெரிந்தெடுக்கலாம். பெருக்கல், கூட்டல் என்பன பற்றிய விளக்கம் உள்ள ஒருவர் நல்ல தீர்வாகத் தீர்வு 3 ஐத் தெரிந்தெடுக்கலாம். அதற்குக் காரணம் கூட்டுவதற்கும் பெருக்குவதற்குமான செய்கைகள் இழிவளவாக இருத்தலாகும்.

### உதாரணம் 2

மாணவன் ஒருவன் தகவல் மற்றும் தொடர்பாடல் தொழினுட்பவியற் பாடத்திற்காகப் பெற்ற புள்ளிகள் 35 இலும் குறைந்ததெனின் அவன் சித்தியடையவில்லையென அல்லது 35 அல்லது அதிலும் கூடியதெனின் அவன் சித்தியடைந்திருப்பதாகக் காட்டல்.



உள்ளீடு : புள்ளிகள்

முறைவழியாக்கம் : பெற்றுள்ள புள்ளிகளை 35 உடன் ஒப்பிடுதல்

தீர்வு 1 : புள்ளிகள் 35 இலும் குறைவெனின்  
பேறு = சித்தியடையவில்லை

அவ்வாறு இல்லாவிட்டால்

பேறு = சித்தி

தீர்வு 2 : புள்ளிகள் 35 அல்லது அதிலும் கூடியனதெனின்

பேறு = சித்தி

அவ்வாறு இல்லாவிட்டால்

பேறு = சித்தியடையவில்லை

வெளியீடு : சித்தி அல்லது சித்தியடையவில்லை

### உதாரணம் 3

இரு எண்களில் பெரிய எண்ணைக் காணல் (பக்கம் 3 இலுள்ள பிரச்சனை 6 ஐப் பார்க்க)

உள்ளிடப்படும் இரு எண்களும்  $n1$ ,  $n2$  எனக் கொள்ளப்படும்.

தீர்வு 1:  $n1$  ஆனது  $n2$  இலும் பெரிதெனின் பெரிய எண்  $n1$  ஆகும்.

$n2$  ஆனது  $n1$  இலும் பெரிதெனின் பெரிய எண்  $n2$  ஆகும்.

தீர்வு 2 :  $n1$  இலிருந்து  $n2$  ஐக் கழிக்குக.

அப்போது கிடைக்கும் பெறுமானம் பூச்சியத்திலும் பெரிதெனின்  $n1$  பெரிய எண்ணாகும்.

அப்போது கிடைக்கும் பெறுமானம் பூச்சியத்திலும் சிறிதெனின்  $n2$  பெரிய எண்ணாகும்.

இவ்வாறு பிரச்சினைக்கு இருக்கும் மாற்றுத் தீர்வுகளிடையே நல்ல தீர்வைத் தெரிந்தெடுத்தல் மிகவும் முக்கியமாகும்.

## 1.2 பிரச்சினைகளைத் தீர்ப்பதற்கு நெறிமுறைகளை (Algorithms)

### உருவாக்கல்

ஒரு பிரச்சினையைத் தீர்ப்பதற்குப் பின்பற்றப்படும் படிமுறைகளைக் காட்டுதல் நெறிமுறை எனப்படும். இங்கு முழுத் திட்டத்துடன் சரியாக ஒரு பிரச்சினைக்குத் தீர்வைக் காணத்தக்க ஒரு நடைமுறையைக் காட்டல் அவசியம்.

## உதாரணம் 1

ஒரு கடிதத்தை அஞ்சலில் இடுவதற்கான ஒரு நெறிமுறையை உருவாக்குவோம்.



- (1) கடிதத்தை எழுதுதல்
- (2) கடிதத்தை மடித்தல்
- (3) கடிதவுறையொன்றில் இடுதல்
- (4) முகவரியை எழுதுதல்
- (5) முத்திரையை ஒட்டுதல்
- (6) அஞ்சலில் இடுதல்

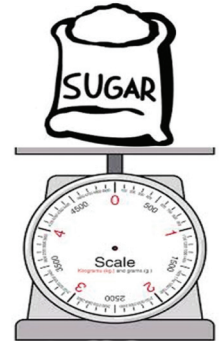
மேற்குறித்த படிமுறைகள் ஒரு கடிதத்தை அஞ்சலில் இடுவதற்கான நெறிமுறை ஆகும். இந்நெறிமுறையில் (1), (2), (3) ஆகிய படிமுறைகளை ஒழுங்கு முறையாகச் செய்தல் வேண்டும். (4), (5) ஆகிய படிமுறைகளை மாற்றலாம். அதற்குக் காரணம் முகவரியை எழுதி முத்திரையை ஒட்டக்கூடியமையாகும். அவ்வாறே முத்திரையை ஒட்டி முகவரியை எழுதலாம்.

அதற்கேற்ப ஒரு நெறிமுறையில் ஒழுங்குமுறையாகச் செயற்படுத்த வேண்டிய படிமுறைகள் இருக்கும் அதேவேளை சில சந்தர்ப்பங்களில் சில படிமுறைகளின் ஒழுங்கு முறையை மாற்றினாலும் அதன் மூலம் முன்வைக்கும் முறைவழியாக்கத்துக்கு தீங்கு ஏற்படமாட்டாது.

## உதாரணம் 2

500 g சீனியைத் தராசின் உதவியுடன் நிறுத்துப் பையில் இடுதல்.

- (1) சீனியைப் பையில் இடுதல்.
- (2) தராசின் மீது வைத்து வாசிப்பை வாசித்தல்.
- (3) சீனி 500 கிராமிலும் குறைந்ததெனில் வாசிப்பு 500 கிராமிற்குச் சமமாகும் வரைக்கும் சீனியைப் பையில் இடுதல்.
- (4) சீனி 500 கிராமிலும் கூடியதெனின் வாசிப்பு 500 கிராமிற்குச் சமமாகும் வரைக்கும் சீனியைப் பையிலிருந்து அகற்றல்.
- (5) வாசிப்பு 500 கிராமாக இருக்கும்போது சீனிப் பையை தராசிலிருந்து அகற்றல்.





## செயற்பாடு



ஓர் ஆரம்பப் பாடசாலையில் 183 மாணவர்கள் உள்ளனர். அவர்களைக் கம்பர், வள்ளுவர், இளங்கோ என மூன்று இல்லங்களாகப் பிரித்து இல்லங்களிடையே விளையாட்டு விழா ஒன்றை நடத்துவதற்கு அதிபர் தீர்மானித்தார். மாணவர்களை இல்லங்களாகப் பிரித்தலைக் காட்டும் நெறிமுறையை உருவாக்குக.

### 1.2.1 நெறிமுறையை விருத்தி செய்வதற்குக் கட்டுப்பாட்டுக் கட்டமைப்பைப் பயன்படுத்தல்

ஒரு நெறிமுறையின் தொழிற்பாடு பற்றிக் கற்கும்போது மூன்று கட்டுப்பாட்டுக் கட்டமைப்புகளை இனங்காணலாம்.

- தொடரி (Sequence)



- தெரிவு (Selection)



- மீள்செயல் (Iteration)



உரு 1.2

## தொடரி

ஒரு நெறிமுறையில் இருக்கும் சில படிமுறைகளை அல்லது எல்லாப் படிமுறைகளையும் ஒரு தொடக்கப் படிமுறையிலிருந்து இறுதிப் படிமுறை வரைக்கும் ஒழுங்கு முறையாக நடைமுறைப்படுத்தல் தொடரி (Sequence) எனப்படும்.

## உதாரணம்

1. ஒரு படிவரிசையில் நடந்து செல்லல்  
ஒரு படிவரிசையில் நடந்து செல்லும்போது படிக்குப் படி ஏறுதல் அல்லது இறங்குதல்
2. நீங்கள் தரம் 1 இலிருந்து தரம் 11 வரைக்கும் சித்தியடைதல்  
பாடசாலையில் தரம் 1 (உட்பட) இலிருந்து தரம் 11 வரைக்கும் தொடர்ச்சியாகக் கல்வி கற்றல்



## செயற்பாடு



ஒரு தொடரியுடன் மூன்று நிகழ்ச்சிகளைக் குறிப்பிடுக.

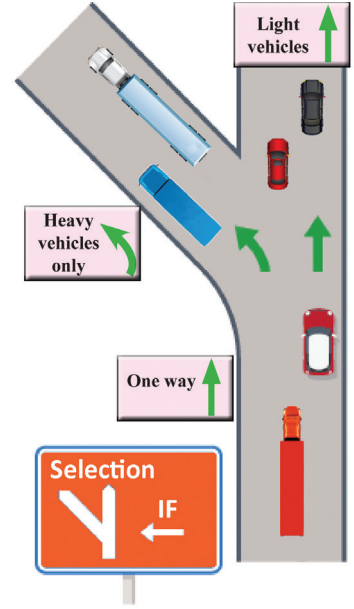


## தெரிவு

ஒரு நெறிமுறையில் உட்பட்ட ஒரு நிபந்தனையைத் திருப்தியாக்குவதற்கு அல்லது திருப்தியாக்காமைக்கேற்பச் செயற்படுத்த வேண்டிய படிமுறைபற்றித் தீர்மானிக்கப்படும் ஒரு சந்தர்ப்பம் தெரிவு (Selection) எனப்படும். இங்கு இரு தெரிவுகள் இருக்கும் அதேவேளை நிபந்தனை திருப்தியாக்கப்படுமாயின் ஒரு தெரிவும் திருப்தியாக்கப்படாவிட்டால் மற்றைய ஒரு தெரிவும் தெரிந்தெடுக்கப்படும்.

### உதாரணங்கள்

1. தரம் 1 இற்கு ஒரு பிள்ளையை அனுமதித்தல்  
அவ்வாண்டில் சனவரி 31 ஆந் திகதி ஒரு பிள்ளையின் வயது 5 இலும் கூடியதெனின் பாடசாலைக்கு அனுமதிக்கலாம்.  
அவ்வாறு இல்லாவிட்டால் பாடசாலைக்கு அனுமதிக்க முடியாது
2. ஒரு பாடத்தில் சித்தியடைதல்  
புள்ளிகள் 35 அல்லது அதிலும் கூடியதெனின் சித்தி அவ்வாறு இல்லாவிட்டால் சித்தியடையாமை.
3. ஒரு புத்தகத்தைக் கொள்வனவு செய்தல்  
புத்தகத்தின் விலைக்குச் சமனான அல்லது அதிலும் கூடிய தொகை உங்களிடம் இருந்தால் புத்தகத்தைக் கொள்வனவு செய்யலாம்.  
அல்லது புத்தகத்தைக் கொள்வனவு செய்ய முடியாது.



உரு 1.3

### செயற்பாடு



1. தெரிவு உள்ள மூன்று நிகழ்ச்சிகளைக் குறிப்பிடுக
2. ஓர் இலங்கைப் பிரசை 18 வயதைப் பூர்த்தி செய்த பின்னர் அவருக்குச் சர்வசன வாக்குரிமை கிடைக்குமெனின், பின்வரும் வெற்றி டங்களுக்குப் பொருத்தமான சொல்லைத் தெரிந்தெடுக்க.  
வயது 18 அல்லது அதிலும் ... (குறைந்தது, கூடியது) எனின்,  
வாக்குரிமையைப் பிரயோகிக்க ... (முடியும், முடியாது)  
அல்லது  
வாக்குரிமையைப் பிரயோகிக்க ... (முடியும், முடியாது)

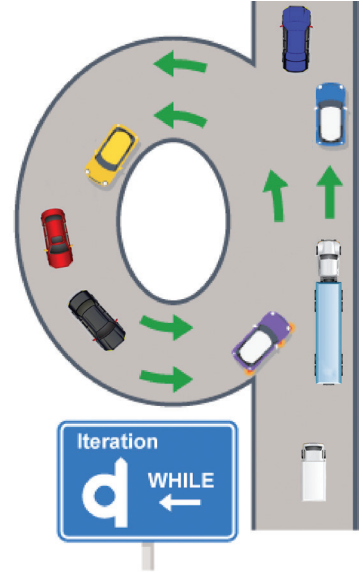
## மீள்செயல் (மறிதரல்)

ஒரு நெறிமுறையில் உட்பட்ட ஒரு படிமுறை அல்லது சில படிமுறைகள் ஒரு நிபந்தனையைத் திருப்தியாக்கும் வரைக்கும் அல்லது திருப்தி செய்யப்பட்டிருக்கும் வரைக்கும் மீண்டும் மீண்டும் நடைபெறுதல் மீள்செயல்/மறிதரல் (Repetition) எனப்படும்.

### உதாரணங்கள்

1. வகுப்புக்குப் பொறுப்பான ஆசிரியர் அல்லது ஆசிரியை மாணவர் இடாப்பில் குறியிடும் முறைவழியாக்கத்தைக் கருதுவோம்.

1. இடாப்பில் முதலில் உள்ள பெயரை வாசித்தல்
2. அம்மாணவர் வந்திருந்தால் 1 எனக் குறிப்பிடுதல்
3. அம்மாணவர் வந்திராவிட்டால் 0 எனக் குறிப்பிடுதல்
4. அடுத்த மாணவனின் பெயரை வாசித்தல்
5. இடாப்பில் உள்ள பெயர்கள் முடியும் வரைக்கும் படிமுறை (2) ஐ அல்லது படிமுறை (3) மற்றும் (4) ஐச் செய்தல்.



உரு 1.4

2. ஒரு பந்தியை வாசித்து அதிலுள்ள சொற்களின் எண்ணிக்கையைக் காணும் முறைவழியாக்கத்தைக் கருதுவோம்.

1. பந்தியில் தொடக்கத்தில் உள்ள சொல்லை வாசித்தல்
2. சொற்களின் எண்ணிக்கை =1
3. அடுத்த சொல்லை வாசித்தல்
4. சொற்களின் எண்ணிக்கையுடன் 1 ஐக் கூட்டல்
5. பந்தி முடிவடையும் வரைக்கும் படிமுறைகள் (3) ஐயும் (4) ஐயும் செய்தல்
6. பந்தியை வாசித்து முடிக்கும்போது சொல் எண்ணிக்கையை எடுத்துரைத்தல்

## செயற்பாடு



1. மீள்செயலுடன் இரு நிகழ்ச்சிகளைப் படிமுறைகளாகக் குறிப்பிடுக.
2. 5 தொடக்கம் 60 வரையுள்ள 5 இன் மடங்குகளை எடுத்துரைப்பதற்கான மீள்செயலுக்குரிய வெற்றிடங்களை நிரப்புக.

படிமுறை I.  $n = 5$

படிமுறை II.  $n$  இன் பெறுமானத்தை எடுத்துரைக்க.

படிமுறை III.  $n$  இன் பெறுமானத்துடன் 5 ஐக் கூட்டுக.

படிமுறை IV.  $n$  இன் பெறுமானம் = 60 வரையுள்ள ..., ... ஆம் இலக்கப் படிமுறைகளைச் செய்க.

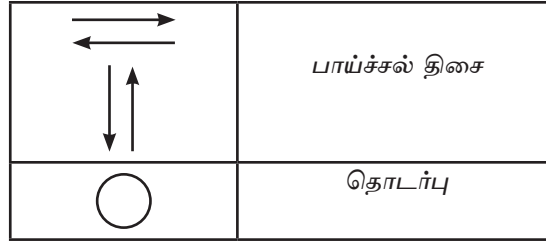
## 1.3 நெறிமுறையை எடுத்துரைப்பதற்குப் பல்வேறு கருவிகளைப் பயன்படுத்தல்

நெறிமுறையை உருவாக்கும்போது எளிதாக எடுத்துரைப்பதற்கும் நெறிமுறையை மிக நன்றாக விளங்கிக்கொள்வதற்கும் பாய்ச்சற் கோட்டுப் படங்கள் (Flowchart), போலிக்குறிமுறை என்னும் முறைகளைப் பயன்படுத்துகின்றோம்

### 1.3.1 பாய்ச்சற் கோட்டுப் படங்கள்

நெறிமுறையை உரு வடிவத்தில் படிமுறைகளாக உருவாக்கும் அல்லது உருவாக் கியுள்ள விதத்தை எடுத்துரைப்பதற்குப் பாய்ச்சற் கோட்டுப்படம் பயன்படுத்தப் படுகின்றது. இங்கு ஒவ்வொரு தொழிற்பாட்டையும் காட்டுவதற்குப் பின்வரும் குறியீடுகள் பயன்படுத்தப்படுகின்றன.

குறியீடு	பயன்பாடு
	தொடக்கம் அல்லது முடிவு
	உள்ளீடு அல்லது வெளியீடு
	முறைவழியாக்கம்
	தீர்மானம்



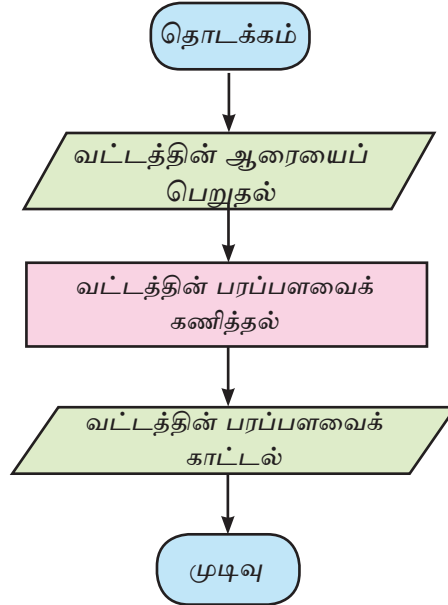
அட்டவணை 1.1 - பாய்ச்சற் கோட்டுப்படக் குறியீடுகள்

### தொடரி உள்ள பாய்ச்சற் கோட்டுப் படங்கள்

தொடக்கப் படிமுறையிலிருந்து இறுதிப் படிமுறை வரைக்கும் உள்ள படிமுறைகள் எல்லாம் ஒன்றிலிருந்தொன்று ஒழுங்குமுறையில் நடைபெறுதல் தொடரியின் இயல்பாகும்.

#### உதாரணம்

ஒரு வட்டத்தின் பரப்பளவைக் கணித்தல்

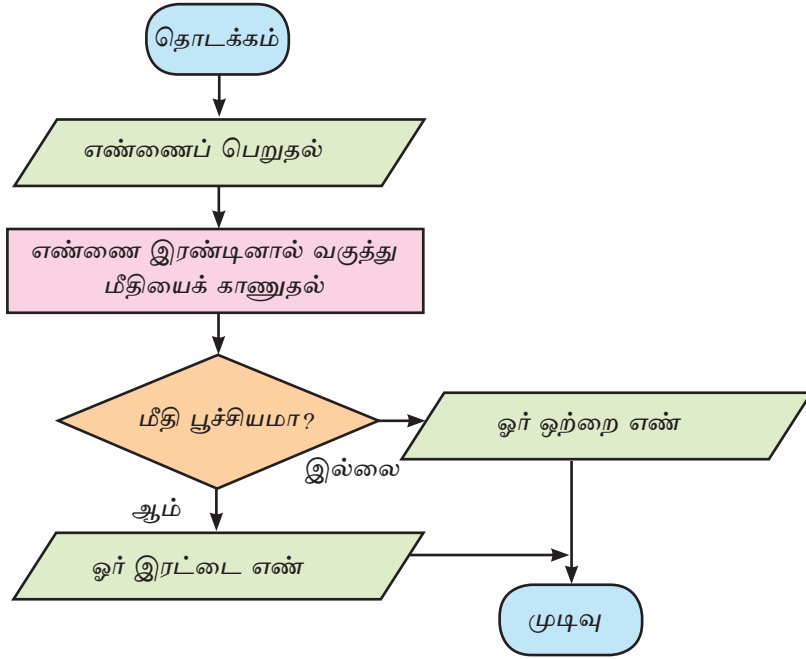


### 1.3.2 தெரிவு உள்ள பாய்ச்சற் கோட்டுப் படங்கள்

ஒரு நிபந்தனை திருப்திசெய்யப்படுவதற்கு அல்லது திருப்தி செய்யப்படாமைக்கு ஏற்பப் பாய்ச்சல் திசை தீர்மானிக்கப்படுதல் தெரிவில் உள்ள சிறப்பியல்பாகும்.

#### உதாரணம்

எண் ஒற்றையா, இரட்டையா எனக் காணல்.

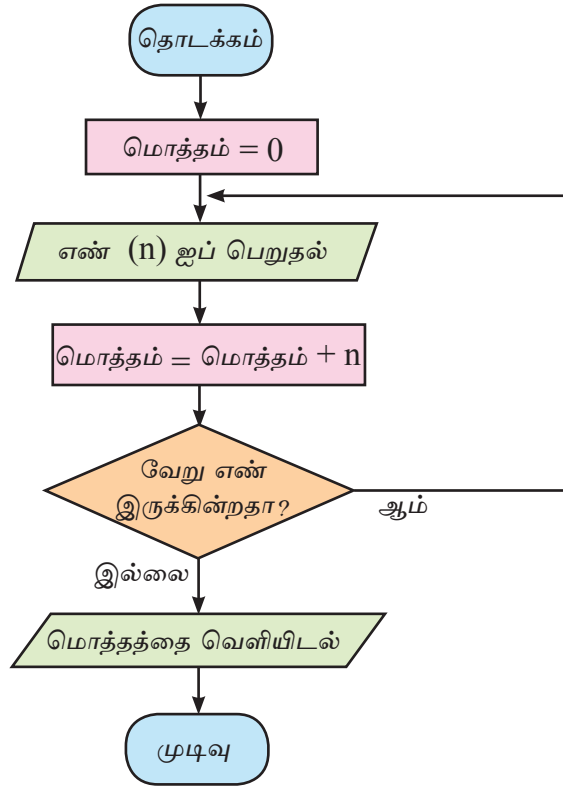


#### மீள்செயல் உள்ள பாய்ச்சற் கோட்டுப்படங்கள்

சில படிமுறைகள் ஒரு நிபந்தனையைத் திருப்திசெய்யும் வரைக்கும் அல்லது திருப்தி செய்திருக்கும் வரைக்கும் மீண்டும் மீண்டும் நடைபெறுதல் இங்கு உள்ள சிறப்பியல்பாகும்.

#### உதாரணம்

ஓர் எண் கூட்டத்தின் கூட்டுத்தொகையைக் காணல்



### செயற்பாடு



பின்வரும் பிரச்சினைகளுக்குப் பாய்ச்சற் கோட்டுப் படங்களை வரைக.

1. ஒரு செவ்வகத்தின் சுற்றளவையும் பரப்பளவையும் காணல்.
2. ஒரு நிறுவகத்தின் தொழிலாளர்களின் அடிப்படைச் சம்பளத்துடன் ரூ. 5000.00 படியைக் கொடுப்பதற்குத் தீர்மானிக்கப்பட்டுள்ளது. இதற்கேற்பப் புதிய சம்பளத்தைக் கணிக்க வேண்டியுள்ளது.
3. ஒரு கடிதத்தைத் தபாலில் இடும்போது அதன் நிறைக்கேற்பக் கட்டணம் அறவிடப்படுகின்றது. நியம நிறைக்குக் குறைந்த அல்லது சமமான நிறையுள்ள கடிதங்களுக்காக குறித்த கட்டணத்தை மாத்திரம் செலுத்த வேண்டும். நியம நிறையிலும் பார்க்கக் கூடிய நிறையுள்ள கடிதங்களுக்காக மேலதிகக் கட்டணத்தைச் செலுத்த வேண்டும்.
4. 7 இன் முதல் 12 மடங்குகளைக் காட்டல்
5. 9 - 10 ஆம் பக்கங்களில் உள்ள உதாரணங்களுக்காகப் பாய்ச்சற் கோட்டுப்படங்களை வரைக.

### 1.3.3 போலிக் குறிமுறைகள்

ஒரு நெறிமுறையை எளிய ஆங்கிலச் சொற்களைப் பயன்படுத்தி எழுத்துருவாகக் காட்டல் போலிக்குறிமுறை (Pseudocode) எனப்படும். இவ்வாறு எழுதப்படும் போலிக்குறிமுறை கணினி மொழியைச் சாராதது. ஒரு போலிக்குறிமுறையை எந்தக் கணினிமொழி அறிவுறுத்தல்களாகவும் மாற்றலாம். ஆகவே போலிக்குறிமுறைகளை எழுதுவதன் மூலம் கணினிச் செய்நிரல்களை உருவாக்குவதை எளிதாக்கலாம்.

நெறிமுறையில் ஒவ்வொரு கோவையும் எளிய ஆங்கிலச் சொற்களினால் காட்டப்படும் விதம்பற்றி ஆராய்ந்து பார்ப்போம்.

தொடக்கம்	Begin
முடிவு	End
உள்ளீடு	Input , Read , Get
வெளியீடு	Output, Display, Show
முறைவழி	Process, Calculate
தெரிவு	IF ... THEN .. .ELSE ... ENDIF
மீள்செய்கை	For – Do
	While – Endwhile
	Repeat – Until

**போலிக்குறிமுறைகளை எழுதுதல்**

**உதாரணம் 1 :-** ஒரு வட்டத்தின் பரப்பளவைக் கணித்தல்

BEGIN

INPUT Radius

CALCULATE Area =  $22/7 \times \text{Radius} \times \text{Radius}$

DISPLAY Area

END.

**உதாரணம் 2 :-** ஓர் எண் ஒற்றையா, இரட்டையா எனக் காணல்

BEGIN

READ number as n

CALCULATE Remainder after number divided by 2

IF Remainder = 0 THEN

DISPLAY “Even number”

ELSE



```

        DISPLAY "Odd number"
    ENDIF
END.

```

**உதாரணம் 3 :-** ஓர் எண் கூட்டத்தின் மொத்தத்தைக் காணல்

```

BEGIN
    Total = 0
    REPEAT
        READ Number as n
        CALCULATE Total = Total + n
    UNTIL numbers are over
    DISPLAY Total
END.

```

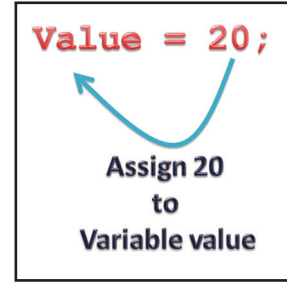
**உதாரணம் 4 :-** 10 எண்களின் மொத்தத்தையும் சராசரியையும் காணல்

```

BEGIN
    Tot = 0
    Avg = 0
    n = 1
    WHILE n <= 10

        READ Num
        CALCULATE Tot = Tot + Num
        n = n + 1
    ENDWHILE
    CALCULATE Avg = Tot/(n-1)
    DISPLAY Tot, Avg
END.

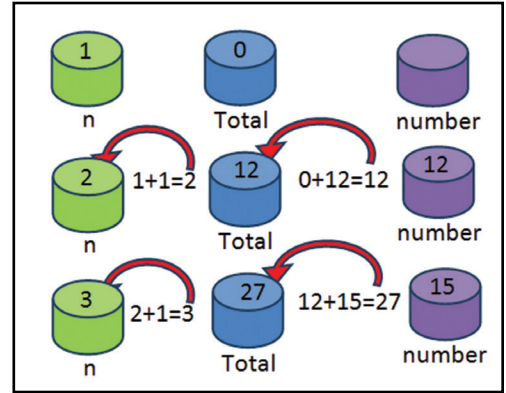
```



மேற்குறித்த போலிக்குறிமுறை பற்றிக் காட்டப்பட்டுள்ள பின்வரும் விடயங்களில் கவனஞ் செலுத்துக.

- Total, Average, number, n ஆகியன மாறிகள் ஆகும்.
- number என்னும் மாறிக்குப் பெறுமானங்களை உள்ளிடும்போது அது தொடர்பாக Total, Average, n ஆகிய மாறிகளின் பெறுமானங்களும் மாறுகின்றன.

- n மாறியின் மூலம் மீள்செயலுக்கு உட்படும் தடவைகளின் எண்ணிக்கை வகைகுறிக்கப்படுகின்றது.
- Total = 0, Average = 0 ஆகியவற்றின் மூலம் அம்மாறிகளின் தொடக்கப் பெறுமானம் 0 ஆக ஒதுக்கப்பட்டுள்ளது.
- number மாறிக்குப் பெறுமானங்களைப் பெறுமுன்பாக Total, Average ஆகிய மாறிகளின் எண் பெறுமானங்கள் 0 என எடுக்கப்பட்டுள்ளன என்பதே இதன் கருத்தாகும்.
- n=1 இன் மூலம் n இன் தொடக்கப் பெறுமானம் 1 என ஒதுக்கப்பட்டுள்ளது.
- o n மாறியின் தொடக்கப் பெறுமானம் 1 என்பதே இதன் கருத்தாகும்.
- n <= 10 ஆனது மீள்செயலைக் கட்டுப்படுத்தும் நிபந்தனை ஆகும்.
- WHILE n <= 10 இன் மூலம் n இன் பெறுமானம் 10 ஆகும் வரைக்கும் மீள்செயலுக்கு உட்பட வேண்டும் எனக் காட்டப்படுகின்றது.
- o n இன் பெறுமானம் 10 அல்லது 10 இற்குக் குறைவாக இருக்கும் சந்தர்ப்பங்களில் மீள்செயல் நடைபெறுகின்றது. அதாவது n <= 10 என்னும் நிபந்தனை உண்மையாக இருக்கும் வரைக்கும் மீள்செயல் நடைபெறுகின்றது. n இன் பெறுமானம் 11 ஆக இருக்கும்போது மீள்செயல் நின்றுவிடுகின்றது. அப்போது நிபந்தனை பொய்யாகும்.
- READ இன் மூலம் number மாறிக்கான ஒரு பெறுமானத்தைப் பெறுதல் காட்டப்படுகின்றது.
- Total + Total = number இன் மூலம் number இற்காகப் பெற்ற பெறுமானம் Total மாறியுடன் கூட்டப்பட்டு அப் பெறுமானம் மறுபடியும் Total மாறிக்கே ஒதுக்கப்படுதல் காட்டப்படுகின்றது
- n=n+1 இன் மூலம் மீள்செயலுக்கு உட்படும் தடவைகளின் எண்ணிக்கை கணிக்கப்படுகின்றது. இங்கு number மாறிக்கு ஒரு பெறுமானத்தைப் பெற்றபின்னர் n இன் பெறுமானத்துடன் 1 ஐக் கூட்டி அப்பெறுமானம் மறுபடியும் n மாறிக்கே ஒதுக்கப்படுகின்றது.
- ENDWHILE இன் மூலம் மீள்செயல் முடிவடையும் எல்லை காட்டப்படுகின்றது. இதற்கேற்ப READ number, Total = Total + number, n = n + 1 ஆகிய கோவைகள் மாத்திரம் n <= 10 என்னும் நிபந்தனை திருப்தியாக இருக்கும் வரைக்கும் மீள்செயலுக்கு உட்படுகின்றன.
- மீள்செயல் நிற்கும்போது n இன் பெறுமானம் 11 ஆகும். அச்சந்தர்ப்பத்தில் நிபந்தனை பொய்யாகும்.



உரு 1.5

- Average = Total/(n-1) இன் மூலம் Total இன் இறுதிப் பெறுமானத்தை n-1 இனால் வகுத்துப் பெறும் பெறுமானம் Average மாறிக்கு ஒதுக்கப்படுகின்றது. அதாவது Average மாறியின் மூலம் சராசரி கணிக்கப்படுகின்றது.
- DISPLAY Total, Average இன் மூலம் உள்ளிடப்பட்ட எண்கள் பத்தின் மொத்தத்தையும் சராசரியையும் வெளியிடுகின்றன.

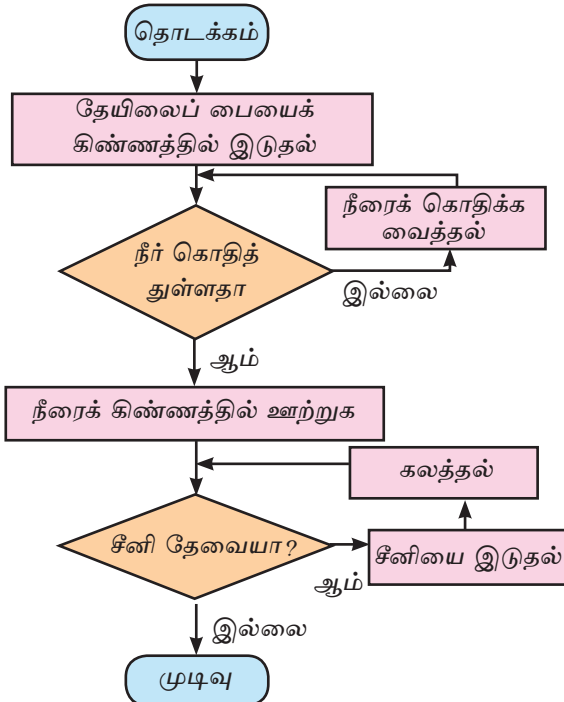
#### அவதானிப்பு



- ஒரு மாறிக்காகப் பெறுமானங்களை ஒதுக்கும்போது முன்னர் இருந்த பெறுமானம் அற்றுப்போகின்றது.
- Total = Total + number என்னும் கோவை தொழிற்படும்போது number மாறிக்கு உள்ள பெறுமானம் Total மாறியில் இருக்கும் பெறுமானத்துடன் கூட்டப்பட்ட பின்னர் கிடைக்கும் பெறுமானம் Total மாறிக்கு ஒதுக்கப்படுகின்றது.
- Total = Total + number ஆனது ஒரு கணிதச் சமன்பாடன்று.

### 1.3.4 பாய்ச்சற் கோட்டுப்படங்களைப் போலிக் குறிமுறைக்கு மாற்றல்

ஒரே நெறிமுறையைப் பாய்ச்சற் கோட்டுப்படத்தைப் போன்று போலிக் குறிமுறையினாலும் வகைகுறிக்கலாம். ஆகவே பாய்ச்சற் கோட்டுப் படத்தைப் போலிக் குறிமுறையாக மாற்றும் விதம்பற்றி ஆராய்வோம்.



BEGIN

Put tea bag in cup

WHILE (not Water boiled)

Boil water

ENDWHILE

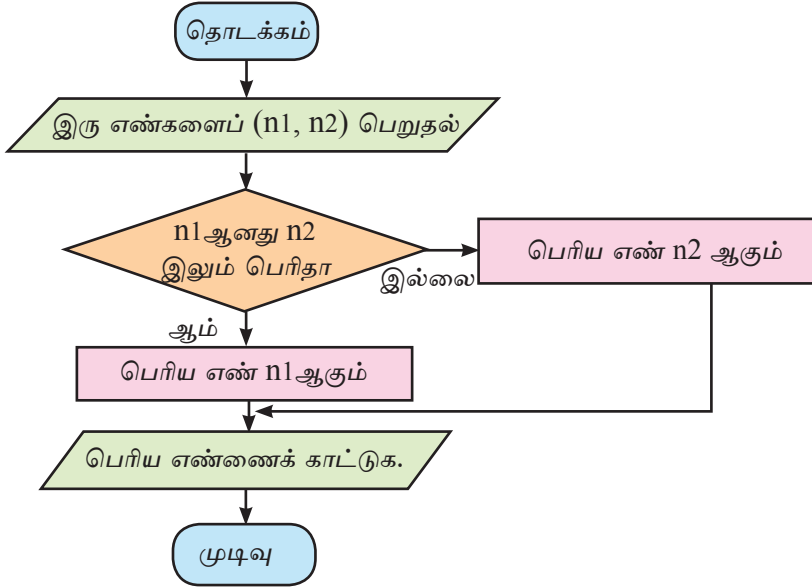
Pour water in cup

WHILE ( sugar needed )  
Add sugar  
Stir tea

ENDWHILE

END

உதாரணம் 1 :- இரு சமமற்ற எண்களிடையே பெரிய எண்ணைக் காணல்



BEGIN

READ n1 , n2

IF n1 > n2 THEN

Large = n1

ELSE

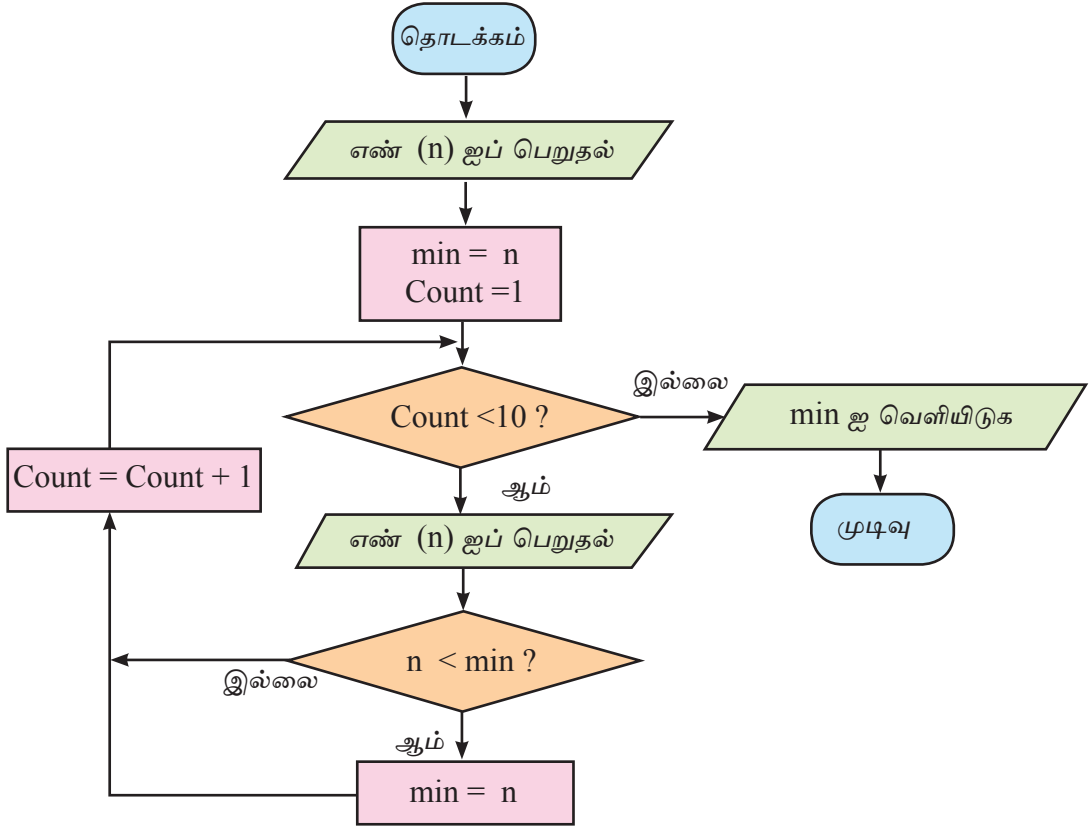
Large = n2

ENDIF

DISPLAY Large

END

உதாரணம் 2 :- பத்து எண்களில் சிறிய எண்ணைக் காணல்.



BEGIN

INPUT Number as n

min = n

Count = 1

WHILE Count<10

INPUT Number as n

IF n < min Then

min = n

ENDIF

Count=Count + 1

ENDWHILE

PRINT min

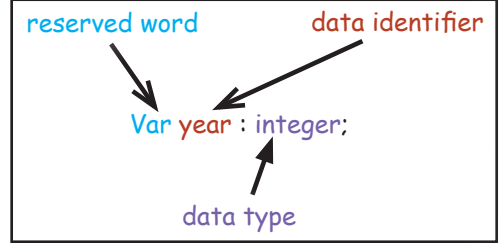
END.

## 1.4 பஸ்கால் செய்நிரல் மொழி

### 1.4.1 அடையாளங்காணி (Identifier)

அடையாளங்காணி என்பது ஒரு மாறியை அல்லது மாறிலியை அல்லது செய்நிரலை அறிமுகஞ்செய்யப் பயன்படுத்தப்படும் பெயர் ஆகும். இந்த அடையாளங்காணியைப் பயன்படுத்துகையில் பின்வரும் பொதுவான விதிகளைப் பின்பற்ற வேண்டும்.

- பஸ்கால் மொழியை அறிமுகஞ் செய்வதற்குப் பயன்படுத்தப்படும் ஒதுக்குசெய் சொற்களை (Reserved words) பயன்படுத்தலாகாது. எந்த வொரு மொழியினதும் ஒதுக்குசெய் சொற்களை அறிமுகஞ்செய்வதற்குப் பயன்படுத்த முடியாது.



- உதாரணம் - BEGIN, END ஆகியன செல்லுபடியாவதில்லை. (உதாரணம் : A-Z, a-z)
- ஆங்கில நெடுங்கணக்கில் உள்ள ஓர் எழுத்துடன் தொடங்க வேண்டும்.
- அடையாளங்காணியின் முதல் எழுத்திற்குப் பின்னர் எழுத்துகளை (a-z, A-Z) அல்லது எண்குறி இலக்கங்களையும் (0-9) கீழ்க்கோட்டையும் ( \_ ) பயன்படுத்தலாம்.  
உதாரணம் - Student\_name
- பஸ்கால் மொழியில் அடையாளங்காணிக்கு எழுத்துப் பேதம் செல்வாக்குச் செலுத்துவதில்லை  
(உதாரணம் - Art, art, ART ஆகியன ஒரே அடையாளங்காணியாகும்)
- சொற்களுக்கிடையே இடைவெளி (Space) இருத்தலாகாது.  
உதாரணம் - Student Name - செல்லுபடியாவதில்லை
- அடையாளங்காணியில் பின்வரும் வரியுருக்கள் இடம்பெறக் கூடாது:  
~ ! @ # \$ % ^ & \* ( ) - + = { } [ ] : ; ' " < > ? , . / | \  
ஆனால் underscore ( \_ ) குறி மாத்திரம் செல்லுபடியாகும்.
- அறிமுகஞ்செய்வதற்கு அர்த்தமுள்ள பெயர்களைப் பிரயோகிப்பதன் மூலம் செய்நிரலை எளிதாக விளங்கிக் கொள்ளலாம்.
- செல்லுபடியான அடையாளங்காணிகளுக்கு உதாரணங்கள் :

Sum, SUM, Total\_Nos, Num1, FirstName, Last\_Name

செல்லுபடியற்ற அடையாளங்காணிகளுக்கு உதாரணங்கள் :

\$75, Average Marks, 9A, Last-name

### 1.4.2 ஒதுக்குசெய் சொற்கள் (Reserved Words)

பஸ்கால் மொழியில் பயன்படுத்தப்படும் ஒதுக்குசெய் சொற்கள் (Reserved Words) பஸ்கால் மொழியை அறிமுகஞ்செய்வதற்குப் பயன்படுத்தப்பட்டுள்ளன. ஆகவே அடையாளங்காணிக்கு ஒதுக்குசெய் சொற்கள் பயன்படுத்தப்படுவதில்லை.

ஒதுக்குசெய் சொற்கள் செய்நிரல் மொழிக்கு மொழி வேறுபடுகின்றன. பஸ்காலில் பயன்படுத்தப்படும் ஒதுக்குசெய் சொற்கள் கீழே காணப்படுகின்றன.

and	exports	mod	shr
asm	file	nil	string
array	for	not	then
begin	function	object	to
case	goto	of	type
const	if	or	unit
constructor	implementation	packed	until
destructor	in	procedure	uses
div	inherited	program	var
do	inline	record	while
downto	interface	repeat	with
else	label	set	xor
end	library	shl	

### 1.4.3 பஸ்கால் மொழியில் பயன்படுத்தப்படும் நியமத் தரவு வகை (Data type)

செய்நிரல் தொழிற்படும்போது உள்ளீடுகளையும் முறைவழியினூடாகக் கணித்த பேறுகளையும் கணினி நினைவகத்தில் வைத்துக்கொள்ள வேண்டும். அதற்குத் தேவையான இடத்தின் அளவு தரவுவகை மீது தீர்மானிக்கப்படும். ஆகவே தரவு வகை (Data type) பற்றிச் செய்நிரலர் அறிந்திருத்தல் மிகவும் முக்கியமானது.

தரவு வகைகளும் அவற்றின் வீச்சுகளும் கீழே காணப்படுகின்றன.

**Integer** - நேர் அல்லது மறை நிறையெண்கள் / முழுவெண்கள்

உதாரணம் - 0, 46, -12



**Real** - நேர் அல்லது மறை மெய்யெண்கள் / தசம எண்கள்

உதாரணம் - 0.0, 25.68

**Boolean**

True அல்லது False

**Char** - சாவிப் பலகையில் உள்ள யாதாயினும் ஒரு வரியுரு

உதாரணம் - 'k', '#', '7'

VARIABLE NAME	VALUE	TYPE
number	- 123	integer
sum	456	integer
character	'B'	char
book	'Mathematics'	string

**String** - யாதாயினும் ஒரு வரியுருத் தொடரி

உதாரணம் - 'ICT', 'programming', 'Sri Lanka'

குறிப்பு : Char, String என்னும் தரவு வகைகளில் ' '(Single quotation) காணப்படுகின்றது.

#### 1.4.4 செய்நிரலில் பயன்படுத்தப்படும் மாறிகளும் (Variable) மாறிலிகளும் (Constant)

**மாறிகள் (Variable)**

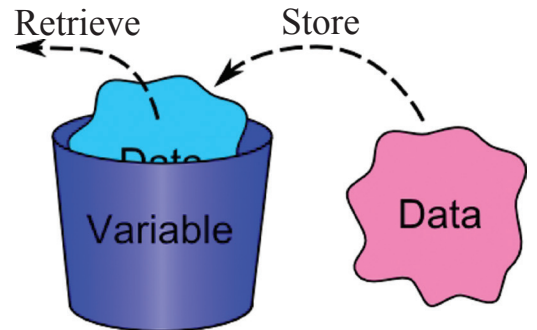
மாறி ஒருவகை அடையாளங்காணி ஆகும்.

செய்நிரலை நடைமுறைப்படுத்தும்போது மாறி அடையாளங்காணிக்கு ஒதுக்கப்பட்ட பெறுமானங்கள் மாற்றமடையும்.

பஸ்கால் மொழியில் மாறியைக் குறிப்பதற்கு "Var" என்னும் ஒதுக்குசெய் சொல் பயன்படுத்தப்படுகின்றது.

உதாரணம்

```
Var count : integer;  
Var a,b : Real;  
Var n1, n2 : integer;  
Avg : real ;  
Pass : boolean;  
Character : char;  
Name,school : String;
```



உரு 1.6

முக்கியம்



ஒரு மாறிக்காக ஒரு பெயர் இருக்கும் அதே வேளை அதில் நிச்சயமான தரவு வகைக்குரிய தரவுகள் தேக்கி வைக்கப்படுகின்றன.

A variable has a name, stores a value of the declared type.

### மாறிலிகள் (Const)

செய்நிரலை நடைமுறைப்படுத்தும்போது அடையாளங்காணிக்கு ஒதுக்கப்பட்ட பெறுமானங்கள் மாறாத அடையாளங்காணி மாறிலி எனப்படும்.

பஸ்கால் மொழியில் மாறிலியை வரையறுப்பதற்கு "const" என்னும் ஒதுக்குசெய் சொல் பயன்படுத்தப்படுகின்றது.

உதாரணம்      Const              max = 100;  
                         Const              pi = 22/7;

### அவதானிப்பு



- செய்நிரலைச் செலுத்தும்போது ஒரு மாறிக்காகப் பல்வேறு பெறுமானங்கள் இருக்கும் அதே வேளை ஒரு மாறிலியின் பெறுமானம் மாறாமல் இருக்கும்

## 1.5 வினைக்குறிகள் / செயலி (Operators)

வினைக்குறிகள் இல்லாமல் கணித்தல், ஒப்பிடுதல், தருக்கக் கோவைகளை உருவாக்கல் ஆகியவற்றைச் செய்யமுடியாது. ஆகவே செய்நிரல்களை எழுதுவதற்கு வினைக்குறிகள் அத்தியாவசியமாகும்.

அடிப்படை வினைக்குறி வகைகள்

### 1. எண்கணித வினைக்குறிகள்

வினைக்குறி	பயன்பாடு	கோவை	பெறு
+	கூட்டல்	6 + 3	9
-	கழித்தல்	7 - 5	2
*	பெருக்கல்	2 * 5	10
/	வகுத்தல்	10/4	2.50
DIV	முழுவெண் வகுத்தல்	20 DIV 6	3
MOD	வகுத்தலின் பின் மீதி	20 MOD 6	2

$$\begin{array}{r} 3 \leftarrow \text{DIV} \\ 6 \overline{) 20} \\ \underline{18} \\ 2 \leftarrow \text{MOD} \end{array}$$

## 2. ஒப்பீட்டு வினைக்குறி

பெறுமானங்களை அல்லது கோவைகளை ஒப்பிடுவதற்கு ஒப்பீட்டு வினைக்குறிகள் பயன்படுத்தப்படுகின்றன. ஒப்பீட்டு வினைக்குறிகள் இடம்பெறும் கோவையின் இறுதிப் பேறு எப்போதும் பூலியன் (Boolean) பெறுமானத்தை எடுக்கின்றது. அதாவது கோவை உண்மை அல்லது பொய்யாகும்.

வினைக்குறி	பயன்பாடு	கோவை	பேறு
>	பெரியது	$7 > 3$	உண்மை
>=	பெரியது அல்லது சமன்	$8 >= 8$	உண்மை
<	சிறியது	$3 < 2$	பொய்
<=	சிறியது அல்லது சமன்	$4 <= 6$	உண்மை
=	சமன்	$3 = 1$	பொய்
<>	சமனற்றது	$2 <> 5$	உண்மை

## 3. தருக்க வினைக்குறி

இரு அல்லது பல கோவைகளை ஒருங்கிணைப்பதற்குத் தருக்க வினைக்குறி (Logical Operator) பயன்படுத்தப்படுகின்றது. இதனைப் பற்றி மேலும் கற்பதற்குத் தரம் 10 இல் நீங்கள் கற்ற அடிப்படை தருக்க வாயில்களைப் பார்க்க.

### i) AND வினைக்குறி

இவ்வினைக்குறி “(முதற் கோவை) AND (இரண்டாம் கோவை)” என்ற அடிப்படையில் பயன்படுத்தப்படுகின்றது. அதில் முதற் கோவையும் இரண்டாம் கோவையும் உண்மையாக அல்லது பொய்யாக இருப்பதற்கேற்ப AND வினைக்குறியினால் பெறப்படும் பேறும் உண்மை அல்லது பொய் ஆகும். பின்வரும் அட்டவணையில் AND வினைக்குறியின் தொழிற்பாடு காணப்படுகின்றது.

முதற் கோவை	இரண்டாம் கோவை	(முதற் கோவை) AND (இரண்டாம் கோவை)
பொய்	பொய்	பொய்
பொய்	உண்மை	பொய்
உண்மை	பொய்	பொய்
உண்மை	உண்மை	உண்மை

- உதாரணம் -**
1. (மழைவீழ்ச்சி  $> 56$ ) AND (வெப்பநிலை  $< 30$ ),
  2. (உயரம்  $> 60$ ) AND (வயது  $< 15$ )
  3.  $(3 \geq 2)$  AND  $(3 < 3)$  என்னும் கோவையைக் கருதுவோம்.  
 $3 \geq 2$  என்னும் கோவை உண்மையாகும்.  $3 < 3$  என்னும் கோவை பொய்யாகும். ஆகவே ஒட்டுமொத்தமான கோவையும் பொய்யாகும்.

**முக்கியம்**



- \* இரு கோவைகளில் குறைந்தபட்சம் ஒரு கோவை பொய்யாக இருக்கும்போது AND வினைக்குறி உள்ள ஒட்டுமொத்தமான கோவையும் பொய்யாகும்.
- \* எல்லாக் கோவைகளும் உண்மையாக இருக்கும்போது மாத்திரம் AND வினைக்குறி உள்ள கோவை உண்மையாகும்.

## ii) OR வினைக்குறி

இவ்வினைக்குறி (முதற் கோவை) OR (இரண்டாம் கோவை) என்ற அடிப்படையில் பயன்படுத்தப்படுகின்றது. அதில் முதற் கோவையும் இரண்டாம் கோவையும் உண்மையாக அல்லது பொய்யாக இருப்பதற்கேற்ப OR வினைக்குறி மூலம் பெறப்படும் பேறும் உண்மை அல்லது பொய் ஆகும். பின்வரும் அட்டவணையில் OR வினைக்குறியின் தொழிற்பாடு காணப்படுகின்றது.

முதற் கோவை	இரண்டாம் கோவை	(முதற் கோவை) OR (இரண்டாம் கோவை)
பொய்	பொய்	பொய்
பொய்	உண்மை	உண்மை
உண்மை	பொய்	உண்மை
உண்மை	உண்மை	உண்மை

- உதாரணம் -**
1. (வெப்பநிலை  $> 30$ ) OR (மழைவீழ்ச்சி  $< 55$ )
  2.  $(3 \geq 2)$  OR  $(3 < 3)$  என்னும் கோவையைக் கருதுவோம்.  
 $3 \geq 2$  என்னும் கோவை உண்மையாகும்.  $3 < 3$  என்னும் கோவை பொய்யாகும். ஆகவே ஒட்டுமொத்தமான கோவை உண்மையாகும்.

**முக்கியம்**



- \* இரு கோவைகளில் குறைந்தபட்சம் ஒரு கோவை உண்மையாக இருக்கும்போது OR வினைக்குறி உள்ள கோவையும் உண்மையாகும்.
- \* எல்லாக் கோவைகளும் பொய்யாக இருக்கும்போது மாத்திரம் OR வினைக்குறி உள்ள கோவை பொய்யாகும்.

### iii) NOT வினைக்குறி

எப்போதும் உண்மையான ஒரு கோவை NOT வினைக்குறி மூலம் பொய்யாகக் காட்டப்படும் அதே வேளை பொய்யான ஒரு கோவை NOT வினைக்குறி மூலம் உண்மையானதாகக் காட்டப்படுகின்றது.

கோவை	NOT (கோவை)
பொய்	உண்மை
உண்மை	பொய்

#### உதாரணம்

1. NOT (வெப்பநிலை > 30)
2. NOT (5 = 5) ஒரு பொய்க் கோவையாகும்.  
5 = 5 என்னும் கோவை உண்மையாகும்.  
ஆகவே NOT(5 = 5) என்னும் கோவை பொய்யெனக் காட்டப்படுகின்றது.

### செய்கை முன்நிகழ்வு/முன்னுரிமை

பஸ்கால் கோவையை நடைமுறைப்படுத்தும்போது பின்வரும் வினைக்குறி முன்நிகழ்வு/முன்னுரிமையின் (Operator Precedence) ஒழுங்குமுறை பின்பற்றப் படுகின்றது.

முன்நிகழ்வு மட்டம்	வினைக்குறி	கூடியது
1	NOT	↑ கூடியது
2	* / DIV MOD AND	
3	+ - OR	
4	= <> < <= > >= குறைந்தது	

பஸ்கால் கோவைகளை மதிப்பிடும் விதம்

உதாரணம் - (1)  
5 + 14 MOD 4  
5 + 2  
7

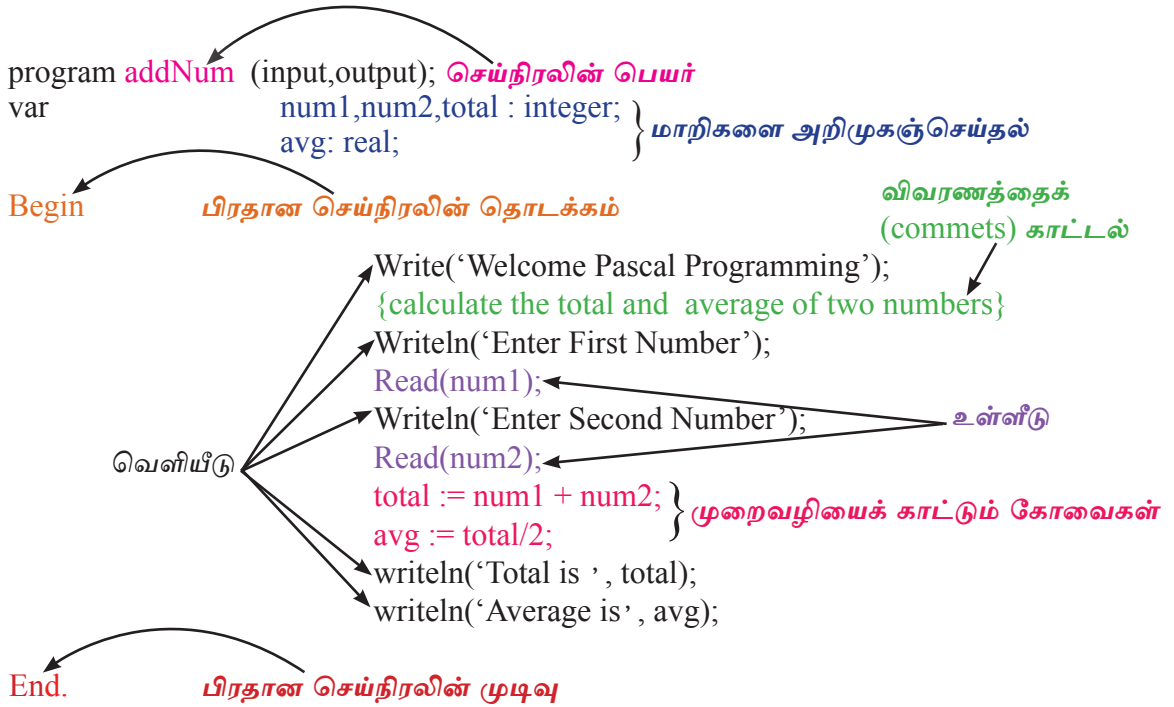
உதாரணம் - (2)  
3 + 7 DIV 2  
3 + 3  
6

உதாரணம் - (3)  
16 / 4 \* 2  
4 \* 2  
8

உதாரணம் - (4)  
NOT (8 MOD 2 > 5)  
NOT (0 > 5)  
NOT(False)  
True

உதாரணம் - (5)  
4 >= 4 AND NOT(7 > 9)  
True AND NOT(False)  
True AND True  
True

ஒரு சாதாரண பஸ்கால் செய்நிரலில் இடம்பெறும் அடிப்படைப் பகுதிகளை இனங்காண்போம்.



குறிப்பு - விவரணத்தைக் காட்டுவதற்கு (\*.....\*) ஐயும் பயன்படுத்தலாம்.

இங்கு "program", "input", "output" ஆகியன ஒதுக்குசெய் சொற்களாகும்.

- "addNum" என்பது ஓர் அடையாளங்காணியாகும். இது செய்நிரலின் பெயராகும். செய்நிரலின் பெயருடன் அடைப்புக்குறிகளினுள்ளே input, output எனக் காட்டல் அத்தியாவசியமன்று.
- உள்ளீட்டுக்காக `read( )`, `readln( )` என்னும் ஒதுக்குசெய் சொற்கள் பயன்படுத்தப் படுகின்றன.
  - `Read(num1);` கோவை மூலம் `num1` ஆன மாறிக்காகத் தரவுகள் உள்ளிடப்படுகின்றன.
  - `readln()` மூலம் தரவுகள் திரையின் ஒரு புதிய நிரையிலிருந்து உள்ளிடப்படுகின்றன.
- வெளியீட்டுக்காக `write( )`, `writeln( )` என்னும் ஒதுக்குசெய் சொற்கள் பயன்படுத்தப்படுகின்றன.
  - `Write ('Welcome to Pascal Programming');` கோவை மூலம் `Welcome to Pascal Programming` என்னும் பாடப் பகுதியை திரையில் வெளியிடலாம்.

- writeln ('Average is ', avg); கோவை மூலம் திரையின் ஒரு புதிய நிரையில் Average is என்னும் பாடப் பகுதியுடன் avg மாறியின் பெறுமானத்தை வெளியிடலாம்.

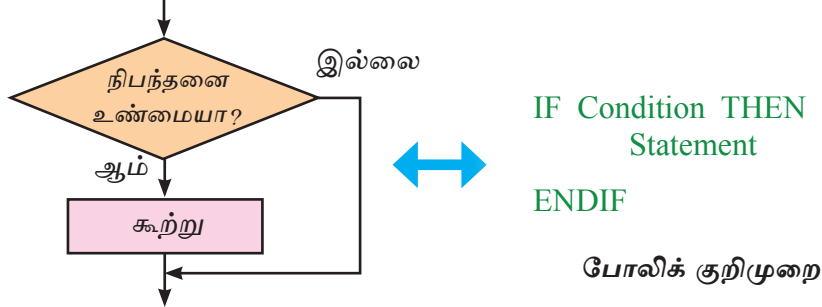
#### பஸ்கால் கூற்றுகளை (Statements) எழுதும்போது

- ஒரு கோவையின் அந்தத்தில் semi-colon (;) இடப்படுகின்றது. semi-colon மூலம் கோவை முடிவடைவதாகக் கூறப் படுகின்றது.
- total:= num1 + num2 ; என்னும் கோவையின் மூலம் num1, num2 என்னும் இரு மாறிகளின் பெறுமானங்களைக் கூட்டி total என்னும் மாறிக்கு ஒப்படைக்கப் படுகின்றது.
- இங்கு பயன்படுத்தப்படும் " :=" ஆனது ஒப்படை வினைக்குறி (Assignment Operator) எனப்படும்.

## 1.6 தெரிவுக் கட்டுப்பாட்டுக் கட்டமைப்புத் தொடர்புபடும் செய்நிரலை அமைத்தல்

### எளிய IF கோவை

தெரிவுக் கட்டுப்பாட்டுக் கட்டமைப்பு மிக எளிய விதத்தில் பின்வருமாறு காட்டப் பட்டுள்ளது. இதில் இரு விதங்கள் உண்டு.

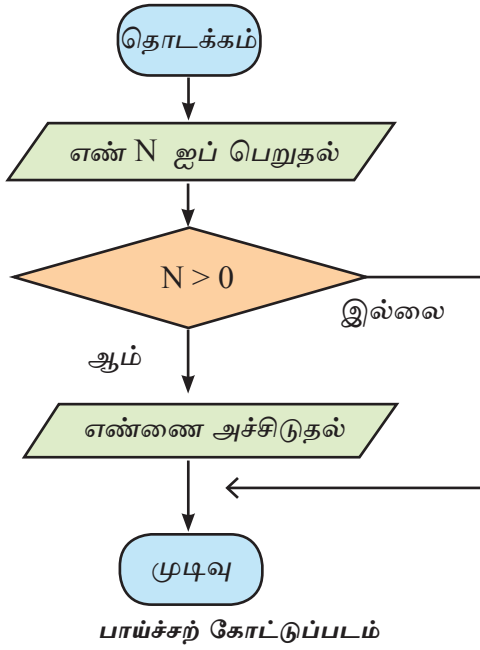


i) IF... THEN.... ENDIF

இங்கு நிபந்தனை திருப்திப்படுத்தப்படின் மாத்திரம் கூற்று நடைமுறைப்படுத்தப்படும்.



உதாரணம் 1 : உள்ளிடப்படும் எண் நேரெனின் மாத்திரம் அந்த எண்ணை அச்சிடுதல்



Begin  
Input N  
IF N > 0  
THEN  
    Print N  
ENDIF  
End.  
போலிக்குறிமுறை

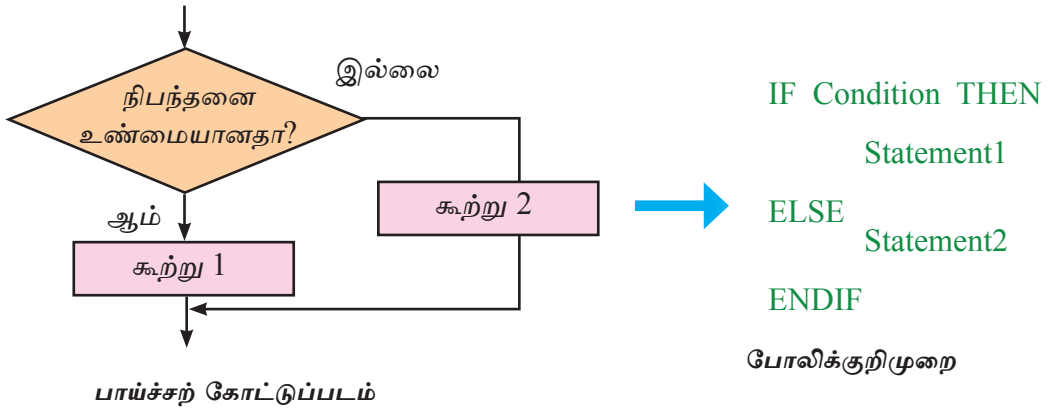
Program positiveNo(input,output);  
Var N : integer;  
Begin  
    Writeln('Enter Number');  
    Read(N);  
    If N > 0 then  
        Writeln('Positive Number');  
    End.  
End.

பஸ்கால் செய்நிரல்

ii) IF... THEN.... ELSE ..... ENDIF

இங்கு நிபந்தனை திருப்தியாக்கப்படுமெனின் கூற்று 1 உம் திருப்தியாக்கப்படாவிடின் கூற்று 2 உம் நடைமுறைப்படுத்தப்படும்.

**உதாரணம் 2 :** சமமற்ற இரு எண்களிடையே பெரிய எண்ணைக் காணல்



```

program LargeNo(input,output);
Var n1, n2, Large: integer;
Begin
    Writeln ('Enter Two Numbers');
    Read(n1,n2);
    If n1 > n2 then
        Large := n1
    Else
        Large := n2;
    Writeln('Large Number is ', Large);
End.
    
```

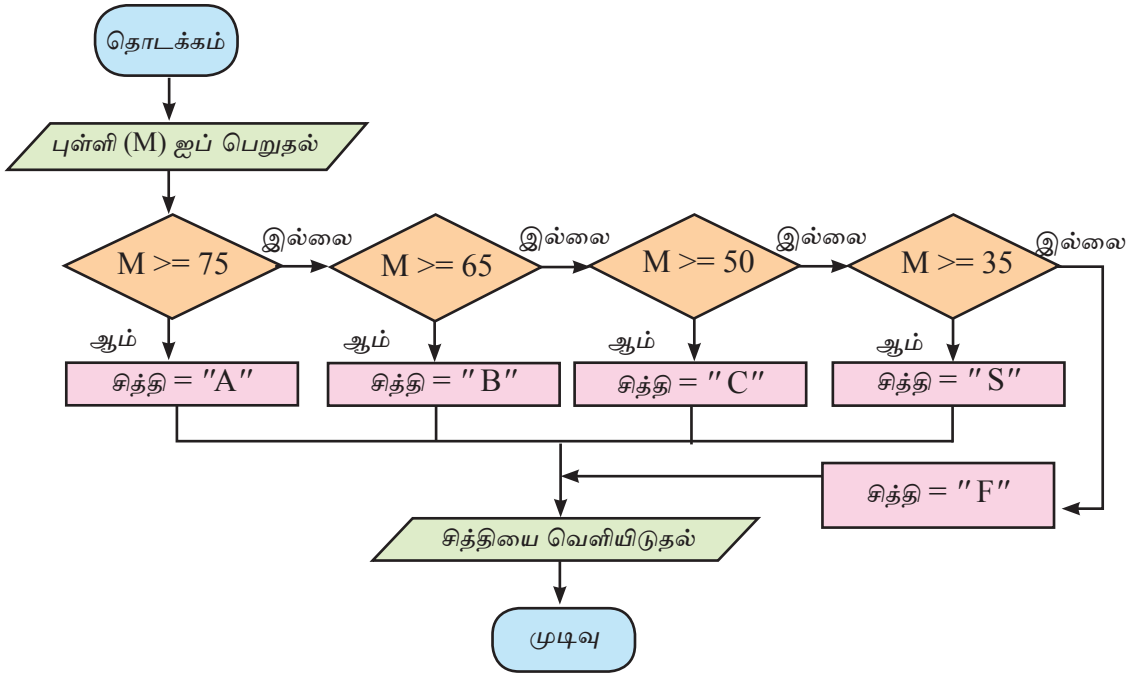
பஸ்கால் செய்நிரல்

### NESTED IF

ஒரு நிபந்தனைக்குப் பின்னர் ஒரு நிபந்தனையாக ஒரு நிபந்தனைக் கூட்டம் வரும் சந்தர்ப்பத்தில் NESTED IF பயன்படுத்தப்படும்.

- ஒரு தனி மாறிக்குப் பல நிபந்தனைகள் இருக்கும்போது NESTED IF ஐப் பயன்படுத்தல்

**உதாரணம் 3:** ஒரு பாடத்திற்கு ஒரு மாணவன் பெற்ற புள்ளிகளை உள்ளிடும்போது அதற்குரிய சித்தியைக் காணல்.



**போலிக்குறிமுறை**

```

Begin
  Input Marks as M
  IF M >= 75 Then
    Grade = "A"
  ELSE
    IF M >= 65 then
      Grade = "B"
    ELSE
      IF M >= 50 then
        Grade = "C"
      ELSE
        IF M >= 35 then
          Grade = "S"
        ELSE
          Grade = "F"
        ENDIF
      ENDIF
    ENDIF
  ENDIF
  Display Grade
End.
  
```

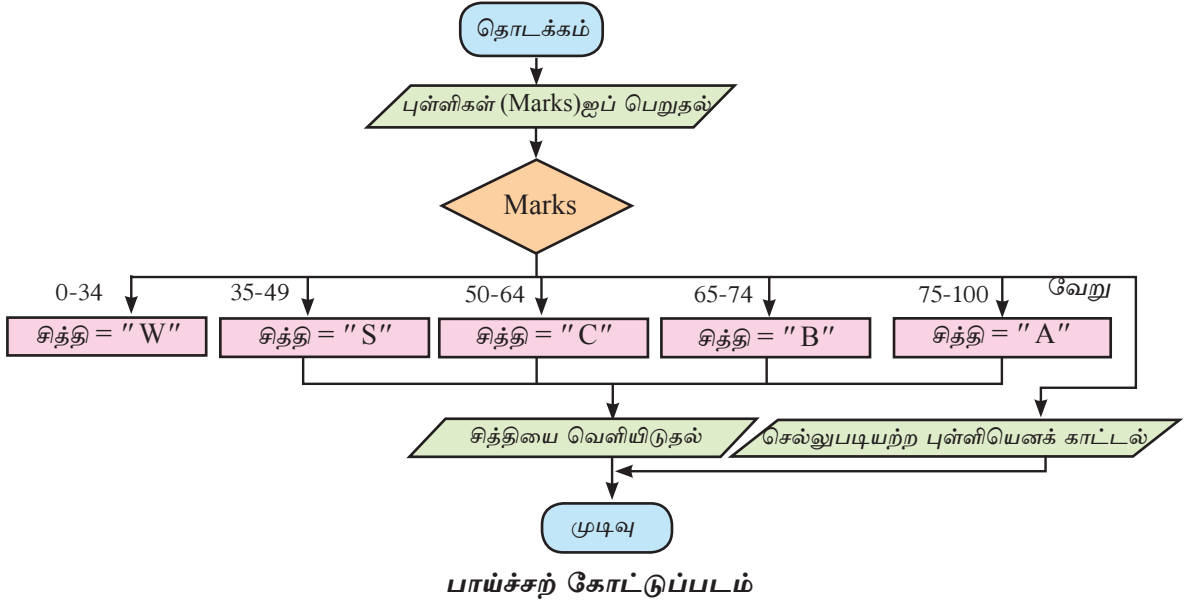
**பஸ்கால் செய்திரல்**

```

program GradeForMarks (input,output);
Var   M: integer;
      Grade: char;
Begin
  Writeln('Enter marks');
  Read(M);
  If M >= 75 then
    Grade := 'A'
  Else
    If M >= 65 then
      Grade := 'B'
    Else
      If M >= 50 then
        Grade := 'C'
      Else
        If M >= 35 then
          Grade := 'S'
        Else
          Grade := 'F';
        End
      End
    End
  End
  Writeln("Grade = ", Grade);
End.
  
```

**தனி மாறுதலுக்குப் பல்நிபந்தனைகள் இருக்கும்போது Case Statement ஐப் பயன்படுத்தல்**

IF ...THEN... ELSE... ENDIF தெரிவுக் கட்டுப்பாட்டுக் கட்டமைப்பை மீண்டும் மீண்டும் பயன்படுத்துவதிலும் பார்க்க எளிதாக Case ஐப் பயன்படுத்தலாம்.



```

program FindGrade(input,output);
var   Marks : integer;
      Grade: char;
Begin
  Writeln('Enter Marks');
  Read(Marks);
  Case Marks of
    0..34 : Grade := 'W';
    35..49 : Grade := 'S';
    50..64 : Grade := 'C';
    65..74 : Grade := 'B';
    75..100 : Grade := 'A';
  Else
    Writeln('Invalid Marks');
  End;
  if (Marks >= 0) AND (Marks <= 100) then
    Writeln('Grade is ', Grade);
End.
  
```

பஸ்கால் செய்நிரல்

## 1.7 மீள்செயல்களைப் பயன்படுத்திச் செய்நிரல்களை எழுதுதல்

மீள்செயல்களின் தடவைகளை நிச்சயமாக அறியும்போது மீள்செயல் கட்டமைப்புகள் பயன்படுத்தப்படும் விதத்தை ஆராய்வோம்.

### i) கட்டமைப்பு (முறை 1) FOR - DO

FOR Variable := Value\_1 TO Value\_2 DO

- இங்கு Variable , Value\_1, Value\_2 ஆகியவற்றின் தரவு வகை முழு எண்களாக (integer) இருத்தல் வேண்டும்.
- மீள்செயலைத் தொடங்குவதற்கு Value\_2 இன் பெறுமானம் Value\_1 இன் பெறுமானத்திலும் பார்க்கக் கூடுதலாக இருத்தல் வேண்டும்.
- மீள்செயல் Value\_1 இல் தொடங்கும் அதேவேளை Value\_2 இல் முடிவடைகின்றது.
- ஆகவே மீள்செயல் நடைபெறும் தடவைகளின் எண்ணிக்கையை நிச்சயமாக அறிந்தால் மாத்திரம் FOR - DO கட்டமைப்பைப் பயன்படுத்தலாம்.

மீள்செயற் கட்டமைப்பு	தொடக்கப் பெறுமானம்	இறுதிப் பெறுமானம்	மீள்செயல் நடைபெறும் தடவைகளின் எண்ணிக்கை
FOR X:= 1 TO 5 DO	1	5	5
FOR X:= 0 TO 4 DO	0	4	5
FOR X:= 5 TO 10 DO	5	10	6

உதாரணம் : 1தொடக்கம் 10 வரையுள்ள பெறுமானங்களை வெளியிடல்

Program print10Nos (input,output);

Var count : integer;

Begin

For count := 1 to 10 do

Writeln(count);

End.

இங்கு count என்னும் மாறியின் பெறுமானத்தை 1 தொடக்கம் 10 வரைக்கும் மாற்றிக்கொண்டு வெளியிடும் அதேவேளை மீள்செயல் பத்துத் தடவைகள் நடைமுறைப்படுத்தப்படும்.

## ii) கட்டமைப்பு (முறை 2) FOR - DO

FOR Variable := Value\_1 DOWNT0 Value\_2 DO

- மீள்செயலைத் தொடங்குவதற்கு Value\_1 இன் பெறுமானம் Value\_2 இன் பெறுமானத்திலும் பார்க்கக் கூடுதலாக இருத்தல் வேண்டும்.
- மீள்செயல் Value\_1 இல் தொடங்கும் அதேவேளை Value\_2 இல்முடிவடைகின்றது.

மீள்செயற் கட்டமைப்பு	தொடக்கப் பெறுமானம்	இறுதிப் பெறுமானம்	மீள்செயல் நடைபெறும் தடவைகளின் எண்ணிக்கை
FOR X:= 10 DOWNT0 5 DO	10	5	6
FOR X:= 4 DOWNT0 0 DO	4	0	5

**உதாரணம் :** 10 தொடக்கம் 1 வரைக்கும் பெறுமானங்களை வெளியிடல்

Program printReverse (input,output);

Var count : integer;

Begin

For count := 10 downto 1 do

Writeln(count);

End.

இங்கு count என்னும் மாறியின் பெறுமானம் 10 தொடக்கம் 1 வரைக்கும் மாறிக் கொண்டு வெளியிடப்படும் அதேவேளை மீள்செயல் பத்துத் தடவைகள் நடைமுறைப்படுத்தப்படுகின்றது.

பத்து எண்களின் மொத்தத்தையும் சராசரியையும் காணல்

```
program total_avg (input,output);
var   I,num,total : integer;
      avg: real;
Begin
  total := 0;
  for I := 1 to 10 do
  begin
    writeln('Enter Number');
    read(num);
    total := total + num;
  end;
  avg := total/I;
  writeln('Total is ', total);
  writeln('Average is ',avg);
end.
```

} மீள்செயலில் ஒரு கூட்டுக் கோவை

**குறிப்பு :** ஒரு கூட்டுக் கோவை begin இற்கும் end ; இற்குமிடையே எழுதப்படுகின்றது. மீள்செயல்களின் தடவைகள் நிச்சயமாகத் தெரியாதவிடத்து While do கட்டமைப்பும் Repeat until கட்டமைப்பும் பயன்படுத்தப்படும்.

#### i) While do கட்டமைப்பு

- ◆ மீள்செயலின் தொடக்கத்திலேயே நிபந்தனை சோதிக்கப்படுகின்றது.
- ◆ நிபந்தனை உண்மையானதெனின் மாத்திரம் மீள்செயல் தொடங்குகின்றது
- ◆ நிபந்தனை பொய்யெனின் ஒருபோதும் மீள்செயல் தொடங்கமாட்டாது.
- ◆ மீள்செயல் முடிவடைவதற்கு நிபந்தனை பொய்யாக இருத்தல் வேண்டும்.
- ◆ மீள்செயல் நடைபெறும் அதேவேளை நிபந்தனை பொய்யன்று எனின், முடிவில் எண்ணிக்கையில் மீள்செயல் (infinite loop) முடிவுறாது தொடரும்.

**உதாரணம் 1 -** while number > 0 do

number மாறியின் பெறுமானம் நேரேனின் மாத்திரம் மீள்செயல் நடைபெறும்

**உதாரணம் 2 -** number := 1;

while number <=10 do

number := number + 1;

- number மாறியின் தொடக்கப் பெறுமானம் 1 ஆகையால் நிபந்தனை உண்மையாகும்.
- ஆகவே மீள்செயல் தொடங்குகின்றது.



- மீள்செயல் நடைபெறும் ஒவ்வொரு தடவையின்போதும் number இன் பெறுமானத்துடன் 1 கூட்டப்படுகின்றது.
- ஆகவே number இன் பெறுமானம் 10 அல்லது 10 இலும் பார்க்கக் குறைவாக இருக்கும்வரை மீள்செயல் நடைபெறுகின்றது.
- number மாறியின் பெறுமானம் 11 ஆக இருக்கும்போது மீள்செயல் நிறுத்தப்படுகின்றது.

## ii) Repeat Until கட்டமைப்பு

- ◆ மீள்செயலின் தொடக்கத்தில் நிபந்தனை சோதிக்கப்படுவதில்லை
- ◆ மீள்செயலாவதற்கு உள்ள கூற்றுகள் ஒரு தடவை நடைமுறைப்படுத்திய பின்னர் நிபந்தனை சோதிக்கப்படுகின்றது
- ◆ நிபந்தனை பொய்யெனின் மாத்திரம் மீள்செயல் தொடங்குகின்றது
- ◆ நிபந்தனை உண்மையாகியதும் மீள்செயல் முடிவடைகின்றது
- ◆ மீள்செயல் நடைபெறும் அதேவேளை நிபந்தனை உண்மையன்று எனின், முடிவில் எண்ணிக்கையில் மீள்செயல் நடைபெறும் (infinite loop)

**உதாரணம் 1 - பஸ்கால் என்னும் சொல் மீள்செயலுக்கு உட்படுத்தல்**

```
count := 0;
Repeat
    writeln ('Pascal');
    count := count+1
Until count > 5;
```

- count மாறியின் தொடக்கப் பெறுமானம் 0 ஆகும்.
- Pascal என்னும் சொல் திரை மீது காட்சிப்படுத்தப்படுகின்றது.
- count மாறியின் பெறுமானத்துடன் 1 கூட்டப்படுகின்றது.
- count மாறியின் பெறுமானம் 5 இலும் பெரிதாவெனச் சோதிக்கப்படுகின்றது.
- இவ்வாறு count மாறியின் பெறுமானம் 5 ஆகும் வரை மீள்செயல் நடைபெறுகின்றது.
- count இன் பெறுமானம் 6 ஆக இருக்கும்போது மீள்செயல் நிறுத்தப்படும்.
- மீள்செயல் நிறுத்தப்படும்போது Pascal என்னும் சொல் திரைமீது 6 தடவை காட்சிப்படுத்தப்படும்.

## உதாரணம் 2

```
sum := 0;
repeat
    sum := sum + 5;
    writeln(sum);
until sum < 50;
```

- ◆ sum மாறியின் தொடக்கப் பெறுமானம் 0 ஆகும்.
- ◆ sum இன் பெறுமானத்துடன் 5 கூட்டப்படுகின்றது.
- ◆ sum இன் பெறுமானம் 5 திரை மீது காட்சிப்படுத்தப்படுகின்றது.
- ◆ sum மாறியின் பெறுமானம் 50 இலும் குறைவானதாவெனச் சோதிக்கப் படுகின்றது.
- ◆ sum < 50 நிபந்தனை திருப்தியாக்கப்படுகின்றது (உண்மையாகும்).
- ◆ ஆகவே மீள்செயல் நின்றிவிடுகின்றது.

## உதாரணம் 3

```
sum := 0;
repeat
    sum := sum + 5;
    writeln(sum);
until sum >= 50;
```

- ◆ sum மாறியின் தொடக்கப் பெறுமானம் 0 ஆகும்.
- ◆ sum இன் பெறுமானத்துடன் 5 கூட்டப்படுகின்றது.
- ◆ sum இன் பெறுமானம் திரை மீது காட்சிப்படுத்தப்படுகின்றது.
- ◆ sum மாறியின் பெறுமானம் 50 இலும் கூடியதா, சமமானதா எனச் சோதிக்கப் படுகின்றது.
- ◆ sum >= 50 நிபந்தனை திருப்தியாக்கப்படும் (உண்மையாக இருக்கும்) வரைக்கும் மீள்செயல் நடைபெறுகின்றது.
- ◆ மீள்செயல் நிற்கும்போது sum இன் பெறுமானம் 50 ஆகும்.
- ◆ 10 தடவை மீள்செயல் நடைபெறுகின்றது.
- ◆ வெளியீடாக 5 தொடக்கம் 50 வரையுள்ள 5 இன் மடங்குகள் பெறப்படும்.

## 1.8 உள்ளமை கட்டுப்பாட்டுக் கட்டமைப்புகளின் தேவைப்பாடு

தெரிவும் மீள்செயலும் ஒன்றிலிருந்தொன்று வேறுபட்ட இரு கட்டுப்பாட்டுக் கட்டமைப்புகளாக இருந்தபோதிலும் பிரச்சினையின் இயல்புக்கேற்ப ஒரு தெரிவுக்குப் பின்னர் ஒரு மீள்செயல் நடைபெறலாம் அல்லது நடைபெறாமல் இருக்கலாம். அவ்வாறே மீள்செயல் நடைபெறும் அதேவேளை ஒரு தெரிவு தொடர்பாக

ஒன்றிலிருந்தொன்று வேறுபட்ட முறைவழியாக்கங்கள் நடைபெறலாம். இத்தகைய நிலைமைகளின் கீழ் செய்நிரலிற்காக உள்ளமை கட்டுபாட்டுக் கட்டமைப்புகளைப் பயன்படுத்த வேண்டும்.

### 1.8.1 தெரிவில் மீள்செயல்களைப் பயன்படுத்தல்

ஒரு தெரிவின் ஒரு நிபந்தனை திருப்தி செய்யப்படுவதற்கு அல்லது திருப்தி செய்யப் படாமைக்கு ஏற்ப ஒரு மீள்செயல் நடைபெறலாம்.

**உதாரணம் -** பயநரின் தெரிவுக்கேற்ப வெளியீட்டில் ஓர் ஏறுவரிசை அல்லது இறங்கு வரிசை எண் தொடர் கிடைத்தல்.

```
program orderNos(input,output);
```

```
var    num:integer;
```

```
      cho:char;
```

```
begin
```

```
    writeln('Select Assending(A) or Desending(D)');
```

```
    read(cho);
```

```
    if cho = 'A' then
```

```
        begin
```

```
            writeln('Asending Order');
```

```
            for num := 1 to 6 do
```

```
                writeln(num);
```

```
            end;
```

```
    if cho = 'D' then
```

```
        begin
```

```
            writeln('Desending Order');
```

```
            for num := 6 downto 1 do
```

```
                writeln(num);
```

```
            end;
```

```
end.
```

### 1.8.2 மீள்செயலில் தெரிவைப் பயன்படுத்தல்

மீள்செயல் நடைபெறும் அதேவேளை ஒரு தெரிவு நடைபெறும் விதத்தைக் கருதுவோம்.

**உதாரணம் :** பயநரால் உள்ளிடப்படும் எண்கள் ஒற்றை எண்களா? இரட்டை எண்களா? என்பதை இனங்காணலும் ஒற்றை, இரட்டை எண் எண்ணிக் கையை வேறுவேறாகக் கணித்தலும்.

```

program rep_sel(input,output);
var num, rem, count, e_count,o_count:integer;
begin
  e_count := 0, o_count := 0;
  for count := 1 to 10 do
    begin
      writeln('Enter Number' );
      read(num);
      rem := num mod 2;
      if rem = 0 then
        begin
          writeln('Even number');
          e_count := e_count + 1;
        end
      else
        begin
          writeln('Odd number');
          o_count := o_count + 1;
        end;
      end;
    end;
  writeln(e_count,'Even Number/s');
  writeln((o_count ,'Odd Number/s');
end.

```

## 1.9 அணியைப் (Array) பயன்படுத்தல்

செய்நிரலில் தரவுகளை நினைவகத்தில் தேக்கிவைப்பதற்கு மாறிகளைக் கட்டாயம் பயன்படுத்த வேண்டும். அவ்வாறே அம்மாறிக்காக உகந்த ஒரு தரவு வகை இருத்தல் வேண்டும். ஒரே தரவு வகைக்குரிய பல தரவுகளைத் தேக்கி வைப்பதற்கு, பெயரில் ஒன்றிலிருந்தொன்று வேறுபட்ட மாறிக் கூட்டம் தேவை.

**உதாரணம் :** 5

முழு எண்களை நினைவகத்தில் தேக்கி வைப்பதற்கு 5 மாறிகள் தேவை. அத்தகைய மாறிகளைப் பயன்படுத்து முன்பாகப் பின்வருமாறு அறிமுகஞ் செய்தல் வேண்டும்.

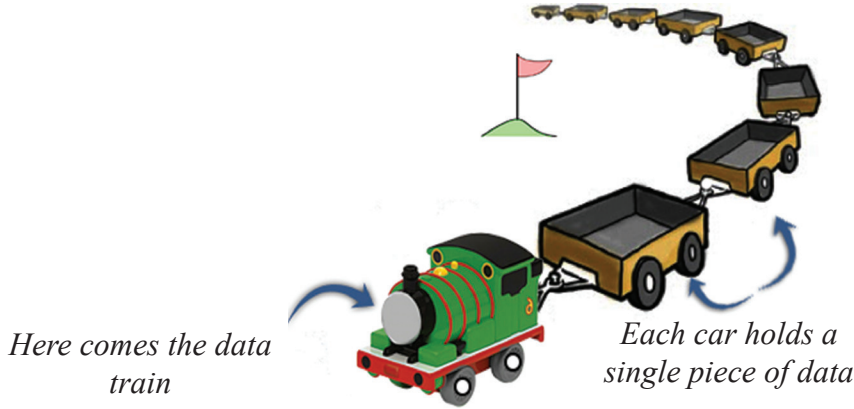
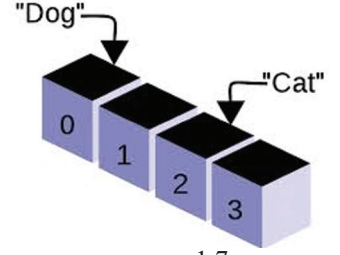
```

Var    p, q, r, s, t : integer; அல்லது
       n1, n2, n3, n4, n5 : real;

```

### 1.9.1 அணிகளைப் பயன்படுத்துவதன் தேவையாடு

ஒரே வகைக்குரிய தரவுகளை ஒரே பெயரைப் பயன்படுத்தி நினைவகத்தில் தேக்கி வைப்பதற்கு அணி பயன்படுத்தப்படுகின்றது. ஆகவே ஒவ்வொரு தரவுக்கும் வேறுவேறாக மாறிப் பெயர்களுக்குப் பதிலாக ஒரு தனி மாறிப் பெயரின் கீழ் தேவையான அளவிற்குத் தரவுகளைத் தேக்கி வைப்பதற்கு அணியைப் பயன்படுத்தலாம்.



உரு 1.8

### 1.9.2 ஒருபரிமாண அணியை வரையறுத்தல்

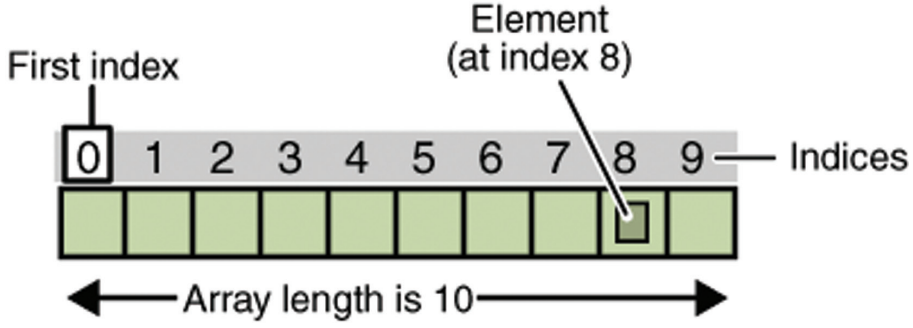
தொடரில் நிச்சய அளவில் ஒரே வகைக்குரிய தரவுகளைத் தேக்கி வைப்பதற்குப் பயன்படுத்தப்படும் தரவுக் கட்டமைப்பு ஒரு பரிமாண அணி ஆகும். ஒரு அணியின் சுட்டி மூலம் அடுத்துள்ள நினைவக வெளிகள் வேறுபடுத்தப்படுகின்றன.

ஒருபரிமாண அணி பின்வருமாறு அறிமுகஞ்செய்யப்படும்.

Var Name\_of\_Array : array [first value .. last value] of data type

உதாரணம் : var marks : array [0..9] of integer;

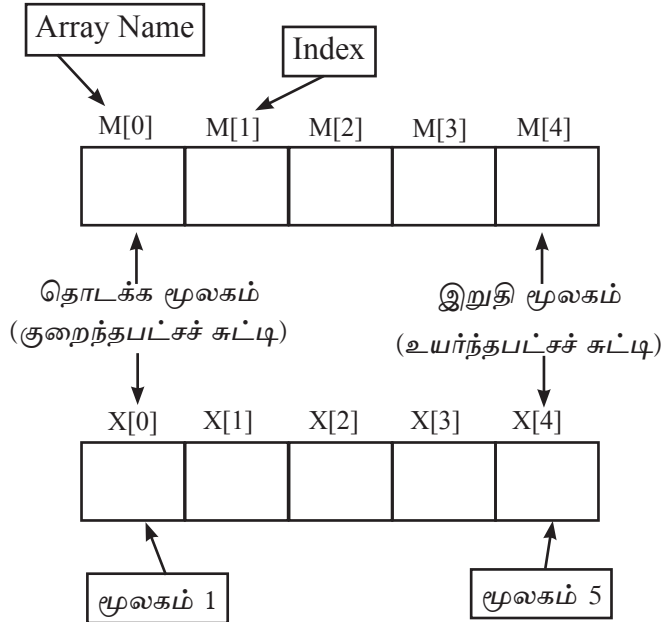
- இதன் மூலம் 10 முழு எண் பெறுமானங்களை உள்ளடக்கத்தக்க marks என்னும் அணி உருவாக்கப்படுகின்றது.



உரு 1.9

### 1.9.3 அணியின் பண்புகள்

- ஓர் அணியின் பகுதிகள் (மூலகங்கள்) அடுத்தடுத்து இருக்கும்.
- ஓர் அணியின் சுட்டி (ஒழுங்குமுறை எண்) அணியின் பெயருடன் அடைப்பினுள்ளே காட்டப்படும்.



#### உதாரணம்

Var M : Array[0..4] of integer; என்னும் அணியின் அளவு 5 ஆகும்.

அது M[0] தொடக்கம் M[4] வரைக்கும் 5 மூலகங்களைக் கொண்டுள்ளது.

அடைப்பினுள்ளே சுட்டி காட்டப்படும்.

அணி அறிமுகஞ் செய்யப்படும் முறைக்கேற்ப அதன் சுட்டி இருக்கும் இடம் வேறுபடும்.

**உதாரணம் :** Var X : Array[1..5] of integer;

ஒரே வகைக்குரிய தரவுகளை மாத்திரம் ஒரு அணியில் தேக்கிவைக்கலாம்.

அணியில் எந்தவொரு மூலகத்திற்கும் எழுமாற்றாகப் பிரவேசிக்கலாம். ஆகவே மீள் செயற் கட்டமைப்பின் மூலம் ஓர் அணியை எளிதாகக் கட்டுப்படுத்தலாம்.

**உதாரணம் :** 40 மாணவர்களின் கணிதப் புள்ளிகளை ஓர் அணியிற் சேர்த்தல்.

```
var    maths : array[0..39] of integer;
      i,marks : integer;
for i := 0 to 39 do
  begin
    writeln('Enter marks');
    read(marks);
    maths[i] := marks;
  end;
```

### 1.9.4 ஓர் அணிக்குப் பெறுமானங்களை வழங்கல்



உரு 1.10

முழுவெண்கள் இடம்பெற்றத்தக்க 5 மூலகங்களைக் கொண்ட num அணியைக் கருதுவோம்.

	num[0]	num[1]	num[2]	num[3]	num[4]
var num : array[0..4] of integer;					
num[0] := 45;					
num[2] := 36; num[4] := 60;	45		36		60
num[1] := num[4] + 15; num[3] := num[0] + num[2];	45	75	36	81	60

### 1.9.5 ஓர் அணியின் பெறுமானங்களைக் காட்சிப்படுத்தல்

ஓர் அணியின் பெறுமானங்களைக் காட்சிப்படுத்தல் அதன் மூலகங்களின் மூலம் நடைபெறுகின்றது.

writeln(num[3]); - 4 ஆம் மூலகத்தைக் காட்சிப்படுத்தல் (81)

writeln(num[1], num[4]); - 2, 5 ஆம் மூலகங்களைக் காட்சிப்படுத்தல் (36, 60)

for x := 0 to 3 do - அணியின் மூலகம் 1 முதல் 4 மூலகங்களை காட்சிப்படுத்தல்  
(45, 75, 36, 81) writeln(num[x]);

for x := 2 to 4 do - அணியின் 3 தொடக்கம் 5 வரையுள்ள 3 மூலகங்களைக்  
காட்சிப்படுத்தல் (36, 81, 60) writeln(num[x]);

for x := 0 to 4 do - அணியின் எல்லா மூலகங்களையும் காட்சிப்படுத்தல் (45,  
75, 36, 81, 60) writeln(num[x]);

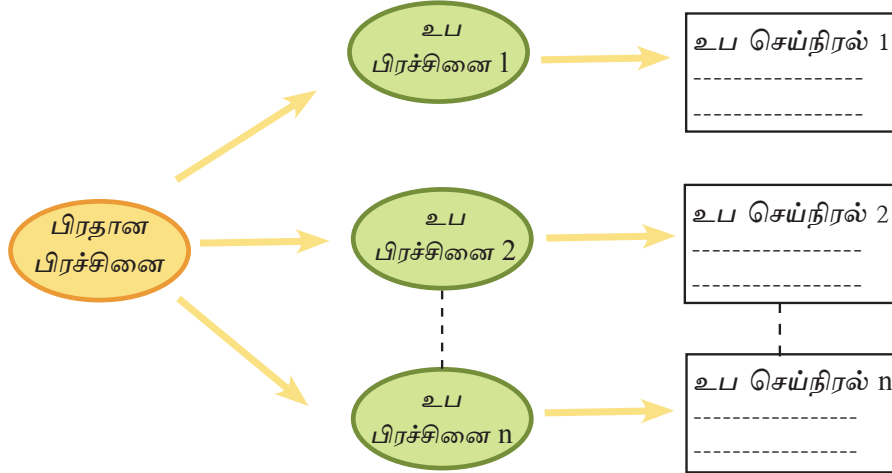
**உதாரணம்** - ஒரு வகுப்பின் 35 மாணவர்கள் தகவல் மற்றும் தொழினுட்பவியல் பாடத்திற்குப் பெற்ற புள்ளிகளை ஒரு அணியில் சேகரித்து கூடுதலான புள்ளியையும் அப்புள்ளிகளின் சராசரிப் பெறுமானத்தையும் கணித்தல்.

```
program ictMarks(input,output);
var marks:array[0..34] of integer;
    i,tot,max:integer;
    avg:real;
begin
    for i := 0 to 34 do
        begin
            writeln('Enter Marks');
            read(marks[i]); (* Read Marks to array *)
            tot := tot + marks[i]; (* Add marks *)
        end;
    avg := tot/35;
    max := marks[0];
    for i := 1 to 34 do
        if marks[i] > max then max := marks[i];
    writeln('Maximum marks = ', max);
    writeln('Average marks = ',avg);
end.
```



## 1.10 உப செய்நிரல்களைப் பயன்படுத்தல்

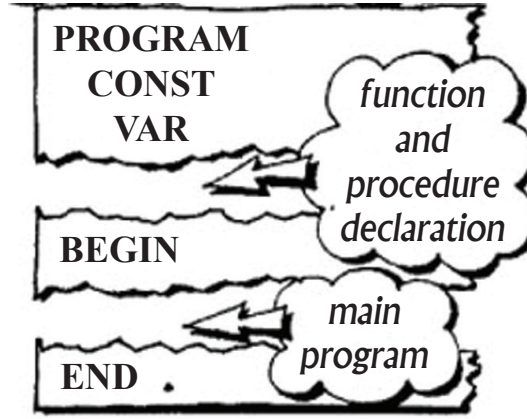
ஒரு செய்நிரலின் முறைவழியாக்கங்களின் அளவு அதிகரிப்பதனுடன் அது சிக்கலானதாகின்றமையால், அதனை வாசித்து விளங்கிக் கொள்ளல், செய்நிரலைப் பேணல் போன்ற பணிகள் கடினமானவையாகும். ஆகவே செய்நிரல்களை எழுதும்போது இயன்ற வரை உபசெய்நிரல்களாக எழுதுதல் ஒரு வெற்றிகரமான முறையாகும்.



### 1.10.1 உப செய்நிரல் வகைகள்

பிரதான செய்நிரலுடன் தொடர்புபட்ட உப செய்நிரல்களின் மூலம் பிரதான செய்நிரலுக்கு வெளியீட்டைத் திரும்ப வழங்கத்தக்க, வழங்க முடியாத இரு வகை உப செய்நிரல்கள் உள்ளன. வெளியீட்டைத் திரும்ப வழங்கத்தக்க உப செய்நிரல்கள் சார்புகள் (Function) எனவும் வெளியீட்டைத் திரும்ப வழங்க முடியாத உப செய்நிரல்களின் நடைமுறை (Procedure) எனவும் அழைக்கப்படும்.

## 1.10.2 உப செய்நிரல்களை அறிமுகஞ்செய்தல்



உரு 1.11

பிரதான செய்நிரலைத் தொடக்குமுன்னர் சார்பையும் நடைமுறையையும் அறிமுகஞ் செய்தல் வேண்டும். பிரதான செய்நிரலில் உப நிரல்களை அழைக்க (Calling the function or Procedure) வேண்டும்.

ஒரு நடைமுறையை அறிமுகஞ்செய்வதற்குரிய சரியான தொடரியல் (Syntax) கீழே காணப்படுகின்றது.

Procedure Name\_of\_Procedure (name\_of\_variable: data type);

**உதாரணம் -** ஒரு வட்டத்தின் பரப்பளவைக் காண்பதற்குரிய நடைமுறை

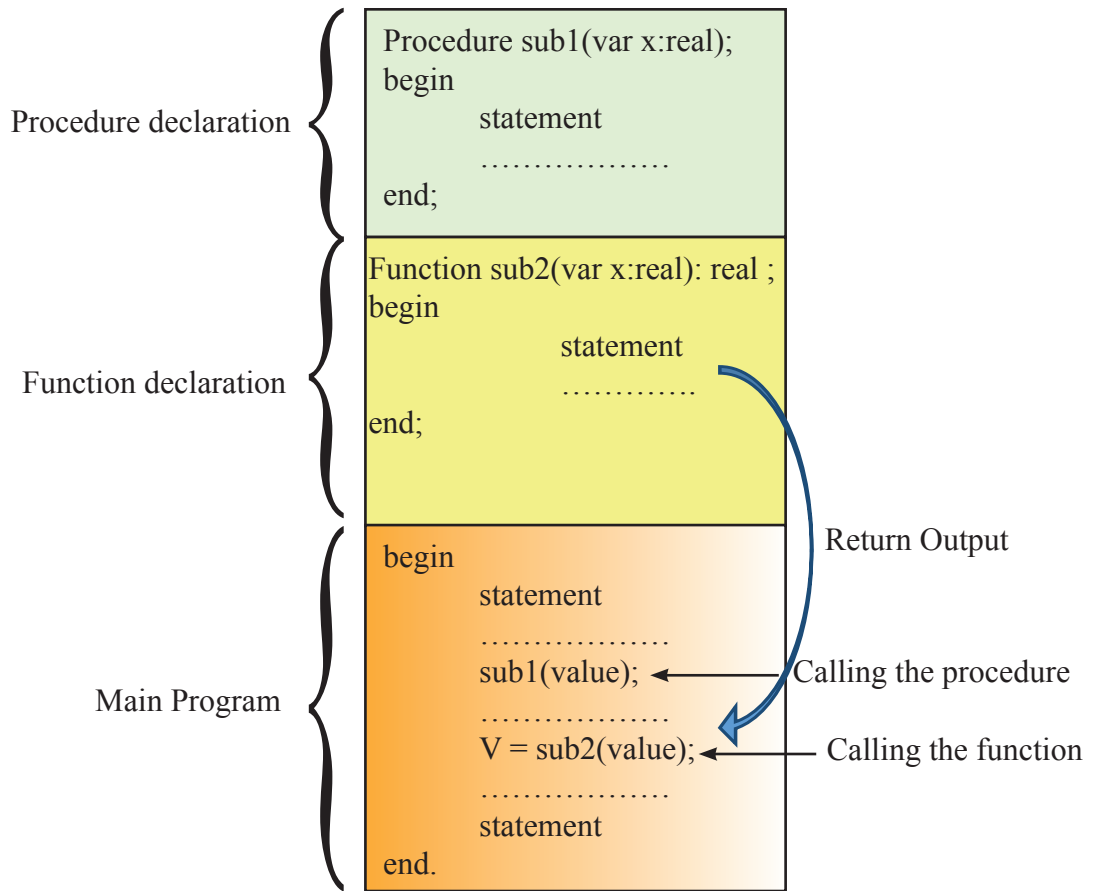
Procedure calculateArea (var radius : real);

ஒரு சார்பை அறிமுகஞ் செய்வதற்குரிய சரியான விதி (Syntax) கீழே காணப்படுகின்றது.

Function Name\_of\_Function(name\_of\_variable : data type) : data type ;

**உதாரணம் -** ஒரு வட்டத்தின் பரப்பளவைக் காண்பதற்குரிய சார்பு

Function calculateArea (var radius : real): real;



**உதாரணம் - ஒரு வட்டத்தின் பரப்பளவையும் பரிதியையும் கணிப்பதற்கான செய்நிரலைக் கருதுவோம்.**

1. நடைமுறையைப் பயன்படுத்தி உருவாக்கப்பட்ட செய்நிரல்

```

program procedure_circle(input,output);
const pie = 22/7;
var radius:real;
procedure getData(var radius: real);
begin
    writeln('Enter Radius');
    read(radius);
end;
procedure ComputeArea(var radius:real);

```

```

    var area:real;
    begin
        area := pie * radius * radius;
        writeln('Area = ',area);
    end;
procedure ComputeCircumference(var radius:real);
    var circum:real;
    begin
        circum := 2 * pie * radius;
        writeln('Circumference = ',circum);
    end;
begin
    getData(radius);
    ComputeCircumference(radius);
    ComputeArea(radius);
end.

```

## 2. சார்பைப் பயன்படுத்தி உருவாக்கப்பட்ட செய்நிரல்

```

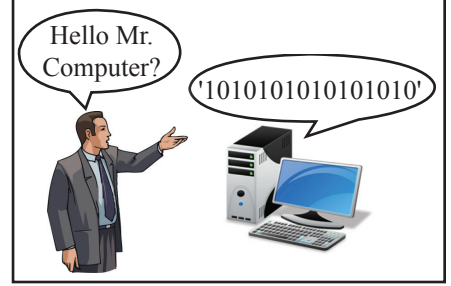
program function_circle(input,output);
const pi = 22/7;
var radius:real;
function ComputeArea(var radius:real):real;
    var area:real;
    begin
        area := pi * radius * radius;
        ComputeArea := area;
    end;
function ComputeCircumference(var radius:real):real;
    var circum:real;
    begin
        circum := 2 * pi * radius;
        ComputeCircumference := circum;
    end;
begin
    writeln('Enter Radius');
    read(radius);
    writeln('Circumference = ',ComputeCircumference(radius));
    writeln('Area = ', ComputeArea(radius));
end.

```

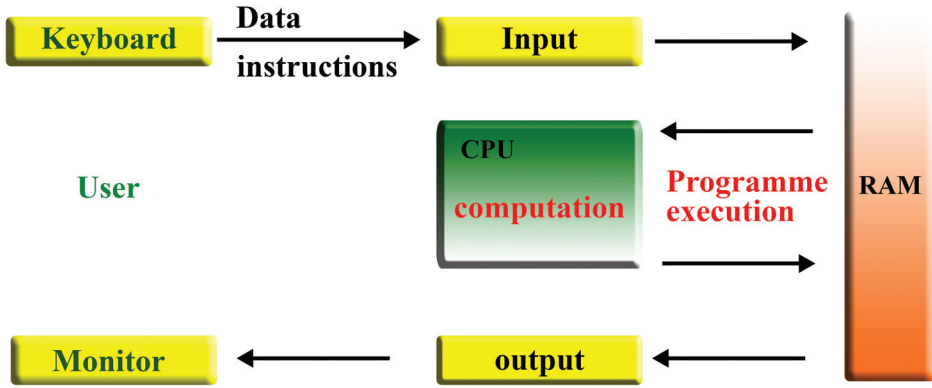
## 1.11 செய்நிரல் மொழிகளின் பரிணாமம்

### 1.11.1 ஒரு செய்நிரல் மொழியின் தேவை

செய்நிரல் என்பது கணினியினால் நிறைவேற்றப்பட வேண்டிய விசேட தொழிலை நிறைவேற்றும் விதத்தைக் காட்டும் அறிவுறுத்தல் தொடரியாகும். இவ்வறிவுறுத்தல்களை வழங்குவதற்குக் கணினி மொழி அவசியமாகும்.



உரு 1.12



கணினியின் செய்நிரல் தொழிற்படும் விதம்

### 1.11.2 கீழ் மட்ட மொழிகள்

#### இயந்திர மொழி

கணினியில் நேரடியாகச் செயற்படுத்தத்தக்க மொழியாகும். அறிவுறுத்தல்களை வழங்குவதற்காக 0, 1 என்னும் துவித எண்கள் (பிற்றுகள்) பயன்படுத்தப்பட்டன. ஆகவே இயந்திர மொழியில் எழுதப்பட்ட ஒரு செய்நிரலை நேரடியாக முறைவழியாக்கத்திற்குச் செலுத்தலாம்.

இயந்திர மொழியில் எழுதப்பட்ட செய்நிரல்

- செய்முறைப்படுத்தல் மிகவும் விரைவாகும்

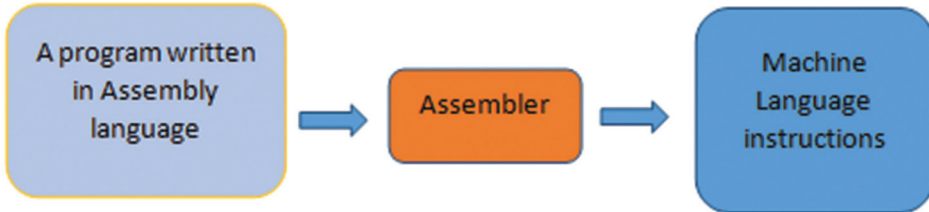
- மொழிபெயர்ப்பு நிகழ்ச்சிநிரல் தேவைப்படாதிருக்கும்
- இயந்திரத்தைச் சார்ந்திருத்தல் (ஒரு கணினிக்கு எழுதப்பட்ட செய்நிரலை வேறொரு கணினியில் இயக்க முடியாதிருக்கும்)
- 0,1 ஆகியவற்றை மாத்திரம் பயன்படுத்தி எழுதப்பட்டிருப்பதனால் விளங்கிக் கொள்ளல் சிக்கலாக இருக்கும்.

### ஒருங்குசேர்ப்பு மொழி

இயந்திர மொழியில் 0, 1 ஆகியவற்றை அடிப்படையாய்க் கொண்டு எழுதப்பட்ட கட்டளைக்குப் பதிலாக எளிய குறியீடுகளில் பெயர்களைப் பயன்படுத்தி ஒருங்கு சேர்ப்பு மொழி அமைக்கப்பட்டுள்ளது.

### ஒருங்குசேர்ப்பு மொழியில் எழுதப்பட்ட செய்நிரல்

- செய்முறைப்படுத்தல் இயந்திர மொழிச் செய்நிரலுடன் ஒப்பிடும்போது விரைவு குறைந்தது.
- ஒருங்குசேர்ப்பி என்னும் மொழிபெயர்ப்பு நிகழ்ச்சித்திட்டத்தின் மூலம் இயந்திரமொழி அறிவுறுத்தலாக மொழிபெயர்க்கப்படுதல் வேண்டும்.
- இயந்திரத்தைச் சார்ந்திருக்கும் (ஒரு கணினிக்கு எழுதப்பட்ட ஒரு செய்நிரலை வேறு கணினியில் இயக்க முடியாதிருக்கும்)
- குறியீடுகள் பயன்படுத்தப்படுகின்றமையால் எளிதாக விளங்கிக்கொள்ளத் தக்கதாக இருக்கும்.



### 1.11.3 உயர் மட்டக் கணினி மொழிகள்

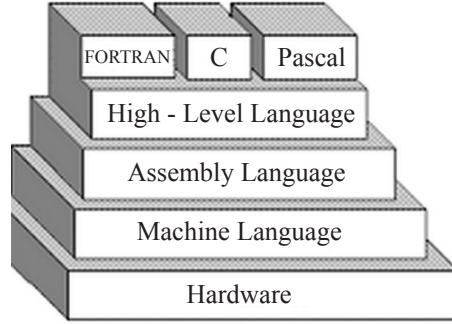
ஆங்கில மொழியில் எளிய சொற்களைப் பயன்படுத்திச் செய்நிரல்களை மிகவும் எளிதாக விளங்கிக் கொள்ளத்தக்கவாறு அமைக்கப்பட்ட கணினி மொழி உயர் மட்டக் கணினி மொழி எனப்படும்.

உயர் மட்டக் கணினி மொழிக்கு உதாரணங்கள்

FORTRAN, BASIC, COBOL, PASCAL, C

உயர் மட்டக் கணினி மொழியில் எழுதப்பட்ட செய்நிரல்களின் இயல்புகள்

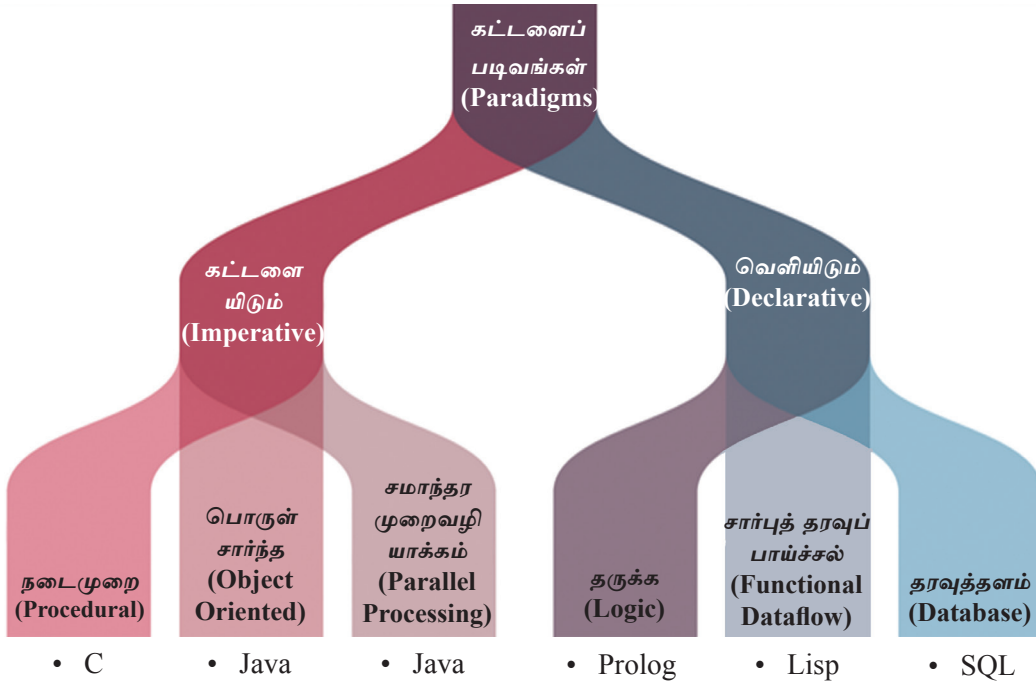
- விளங்கிக் கொள்ளல் எளிதாகும்.
- இயக்கு முன்பாக இயந்திர மொழி அறிவுறுத்தல்களாக மாற்ற வேண்டும்.
- இயந்திரத்தைச் சார்ந்த மொழியன்று.



#### 1.11.4 செய்நிரல் மொழி வகைகள்

செய்நிரலாக்கம் என்பது ஒரு குறித்த தொழிலைச் செய்யும் விதம் பற்றிக் கணினிக்கு அறிவுறுத்தல்களை அளிப்பதற்குக் கணினிச் செய்நிரலரினால் செய்யப்படும் ஆக்கபூர்வமான முறைவழியாகும். ஒரு குறித்த பிரச்சினைக்கு ஒரு தீர்வைத் தருவதற்கு எந்தத் தொழிலைச் செய்தல் வேண்டும் என்பது பற்றிக் கணினிக்கு அறிவுறுத்தலை அளிப்பதற்குத் தயாரித்த அறிவுறுத்தல் தொகுதி செய்நிரல் எனப்படும்.

செய்நிரல் முறைவழிக்குப் பிரவேசிப்பதற்குப் பல மாற்றுப் பெறுவழிகள் உள்ளன. அவை செய்நிரல் கட்டளைப் படிவங்கள் (Paradigms) எனப்படும். செய்நிரல்களைப் பயன்படுத்தி விசேட பிரச்சினைகளுக்குத் தீர்வுகளை உருவாக்குவதற்கு அடிப்படையில் வேறுபட்ட பெறுவழிகள் பல்வேறு கட்டளைப் படிவங்களின் மூலம் குறிக்கப்படுகின்றன. பெரும்பாலான செய்நிரல் மொழிகள் ஒரு கட்டளைப்படிவ வகையின் கீழ் கணிக்கப்படுகின்ற போதிலும் சில மொழிகளில் பல்வேறு கட்டளைப் படிவங்களுக்குரிய மூலத்துவங்கள் இருப்பதைக் காணலாம்.



உரு 1.13  
செய்நிரல் கட்டளைப் படிவம்

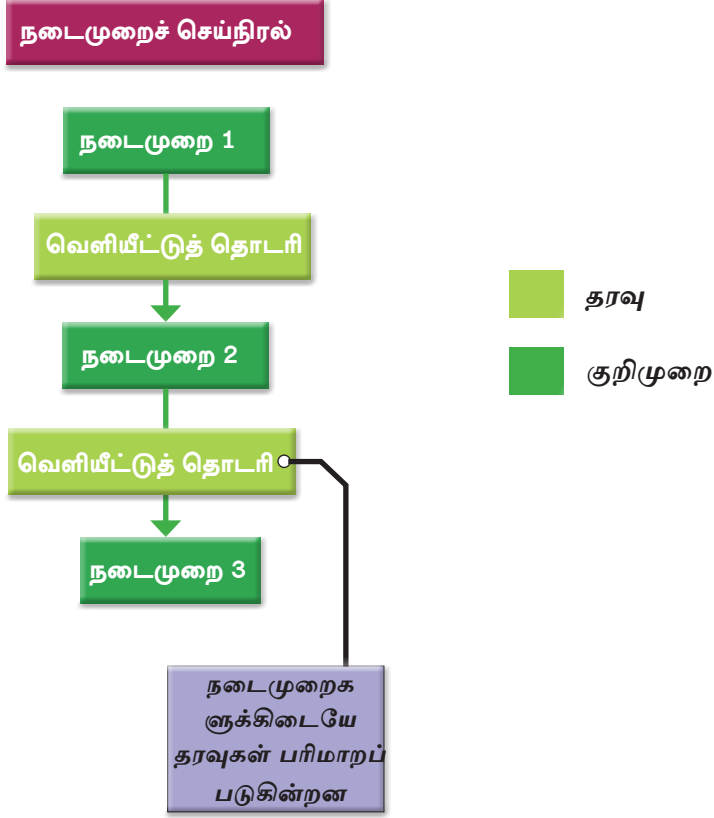
கட்டளையிடும்	வெளியிடும்		பொருள் சார்
	சார்புச் செய்நிரல்	தருக்கச் செய்நிரல்	
Algol Cobol PL/1 Ada C Modula - 3	Lisp Haskell ML Miranda APL	Prolog	Smalltalk Simula C++ Java

இற்றைவரைக்கும் பல செய்நிரல் மொழிகள் உருவாகியுள்ளன. அவை காலத்துடன் மதிப்பீட்டுக்கு உட்படுத்தப்படும்போது பின்வருமாறு மாற்றங்கள் இருப்பதைக் காணலாம்.

நடைமுறைச் செய்நிரலிற்கும் வெளியீட்டுச் செய்நிரலிற்குமிடையே உள்ள வித்தியாசம்

நடைமுறை மொழி என்பது மிகச் சிறந்த கட்டமைப்புப் படிமுறைத் தொகுதியையும் நடைமுறைகளையும் கொண்ட கணினிச் செய்நிரல் மொழி வகை ஆகும். அதில் ஒழுங்கான படிமுறைகளைக் கொண்ட கூற்றுகள் இடம்பெறுகின்றன.





**நீங்கள் கற்ற பஸ்கால் செய்நிரலில் நடைமுறைச் செய்நிரலின் இயல்புகள் உள்ளன.**

வெளியீட்டுச் செய்நிரல் என்பது கட்டுப்பாட்டுப் பாய்ச்சல் விவரணம் இல்லாமல் கணிப்பதன் தருக்கக் கொள்கையைக் காட்டும் கணினிச் செய்நிரல்களின் கட்டமைப்பும் மூலகங்களும் உருவாக்கப்பட்டுள்ள விதமாகும். இந்த ஆக்கபூர்வமான விதம் பயன்படுத்தப்படும் பல நிர்மாணிப்புகளின் மூலம் பக்க விளைவுகளை இழிவள வாக்குவதற்கு அல்லது நீக்குவதற்கு முயலப்படுகின்றது. முதல் செய்நிரல் மொழிகள் போல பிரச்சினையைத் தீர்ப்பதற்கு எங்ஙனம் செயற்படுதல் வேண்டும் என்பதை விவரிப்பதிலும் பார்க்கப் பிரச்சினையைத் தீர்ப்பதற்கு எச்செய்நிரலைத் தயாரிக்க வேண்டுமென விவரிப்பதன் மூலம் இது செய்யப்படுகின்றது. எவ்வாறாயினும் அதன் மூலம் எவ்வாறு நடைபெறுகின்றது என்பது விவரிக்கப்படுவதில்லை அதாவது பிரச்சினை எது என்பது பற்றி மாத்திரம் கணினி அறிவிக்கும் அதேவேளை அது தீர்க்கப்படும் விதம் பற்றி அறிவிப்பதில்லை. உள்ளிடப்படும் பிரச்சினைக்குத் தேவையான தீர்வுகள் கணினி மூலம் கண்டுபிடிக்கப்படுகின்றன. இது விவரமான படிமுறையாக நெறிமுறைச் செயலுக்கு வரும், நடைமுறைச் செய்நிரலுக்கு எதிரானது. வெளியீட்டுச் செய்நிரல் செயற்கை நுண்ணறிவை அடிப்படையாய்க் கொண்டது.



### நடைமுறைக் கட்டளைப் படிவம் (Procedural Paradigm)

தீர்வைக் காணும் விதத்தைக் கூறுதல்  
Saying how you achieve it



1. சதுரமுகி A யை வைக்க
2. சதுரமுகி A மீது சதுரமுகி B யை வைக்க.
3. சதுரமுகி C யைச் சதுரமுகி B மீது வைக்க.

### வெளியீட்டுக் கட்டளைப் படிவம் (Declarative Paradigm)

தேவையானவற்றைக் கூறுதல்  
Saying what you want



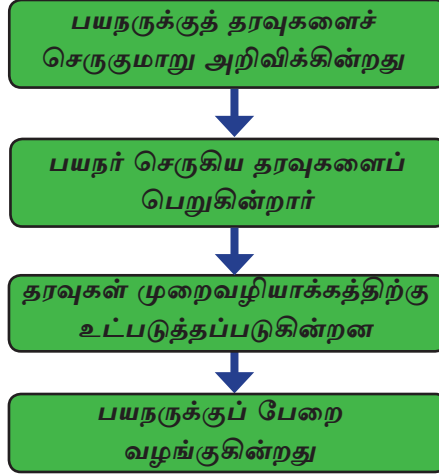
3 சதுரங்களைக் கொண்ட தூண்

கட்டமைப்புச் செய்நிரலையும் பொருள் சார்ந்த செய்நிரலையும் ஒப்பிடுதல்

கட்டமைப்புச் செய்நிரல் (Structured) என்பது தருக்கச் செய்நிரல் கட்டளைப்படிவமாக இருக்கும் அதேவேளை அது பொருள் சார்ந்த செய்நிரலின் முற்பட்ட சந்தர்ப்பமாகும். கட்டமைப்புச் செய்நிரற் கட்டளைப் படிவத்தின் மூலம் செய்நிரலை விளங்கிக் கொள்வதற்கும் நவீனமயமாதலுக்கும் வசதிகள் செய்யப்படுகின்றன. தொகுதி கூறு உபதொகுதிகளாகப் பிரிக்கப்பட்டிருக்கும் அதேவேளை மேலிருந்து கீழுக்குப் பெறுவழிகள் உள்ளன.

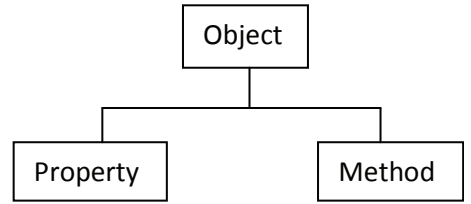
செய்நிரலின் குறிமுறைகளுக்குத் தருக்கக் கட்டளையைக் கையளிக்கும் தொழினுட்பம் இங்கு உண்டு. ஆகவே, அதில் முறைவழி எளிதாக்கப்படுகின்றது. பெரிய செய்நிரல் சிறிய அலகுகளாகப் பிரித்து வேறாக்கப்பட்டிருக்கும் அதேவேளை அப்பகுதிகள் ஒழுங்காக ஒன்றோடொன்று பிணைக்கப்பட்டிருக்கும்.

கட்டமைப்புச் செய்நிரல்

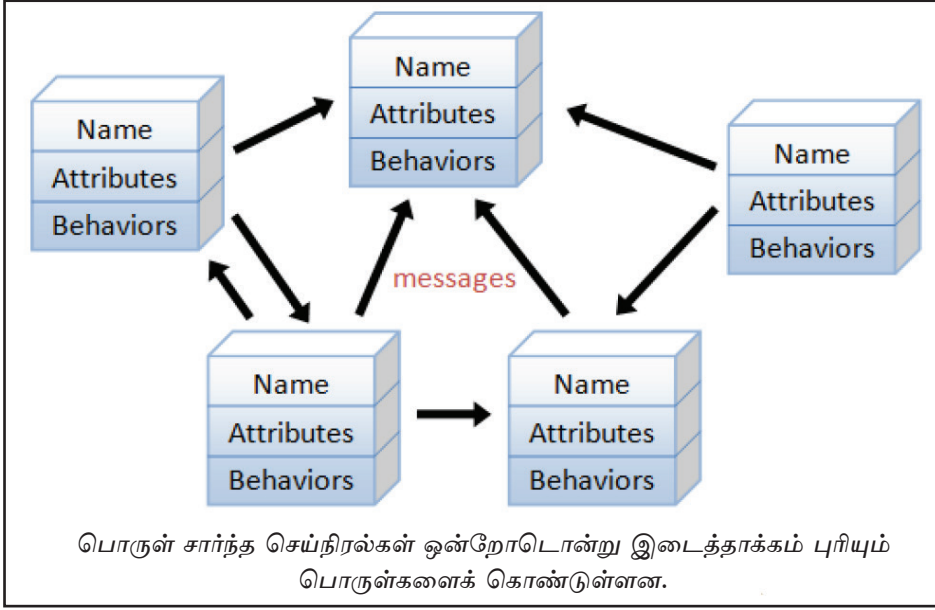


நீங்கள் கற்ற பஸ்கால் செய்நிரலில் கட்டமைப்புச் செய்நிரல் இயல்புகளும் உள்ளன

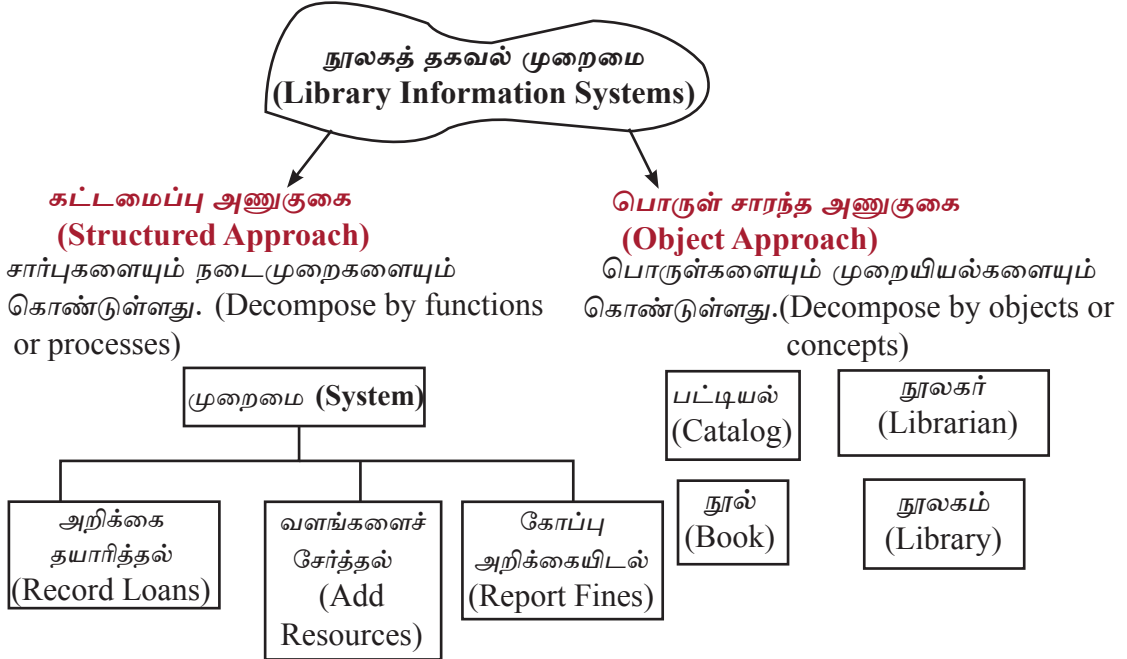
பொருள் சார்ந்த கணினிச் செய்நிரல் என்பது பொருள்கள் (Objects) என்னும் எண்ணக் கருவை அடிப்படையாய்க் கொண்ட செய்நிரற் கட்டளைப்படிவம் ஆகும். பொருள் என்பது தரவுகள் அடங்கும் தரவுக் கட்டமைப்புகள் (Data Structures) ஆகும். இத்தரவுக் கட்டமைப்புகள் உபபண்புகள் எனப்படும் புலங்களின் வடிவத்தில் (Attributes) இருக்கின்றன. அத்தரவுகளைக் கையாள்வதற்கு முறைகள் (Form of fields) எனப்படும். நடைமுறை வடிவத்தில் உள்ள குறிமுறை உள்ளது.

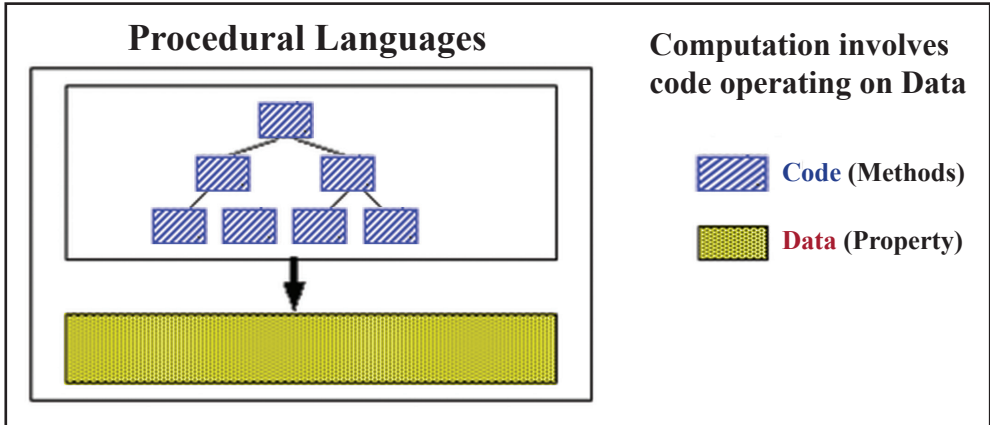
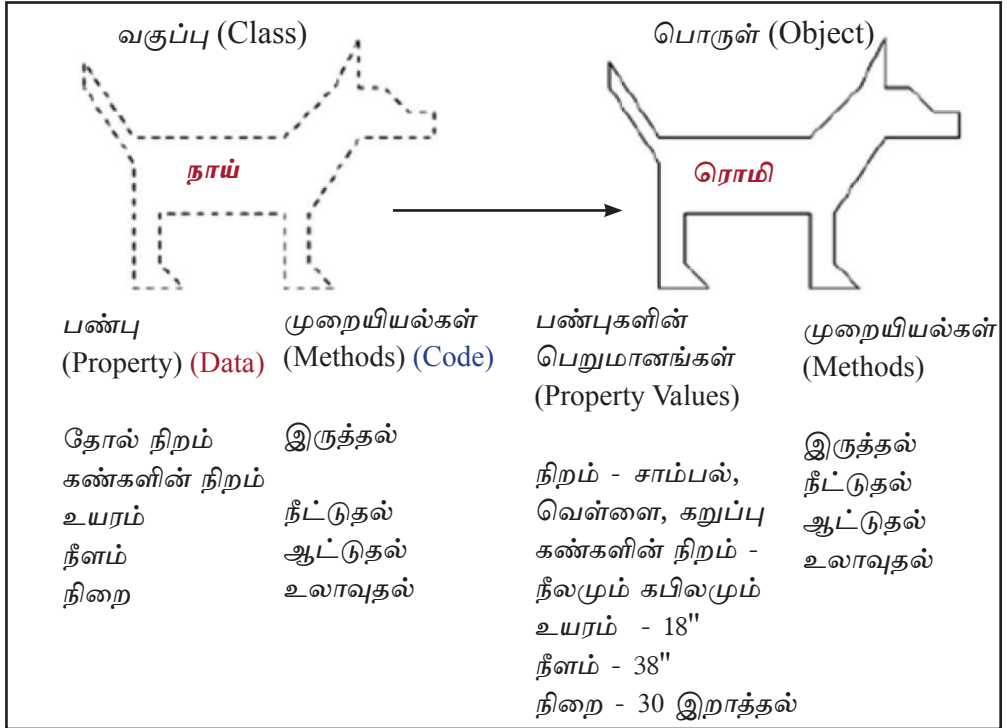


பொருள் சார்ந்த செய்நிரலில் ஒரு பொருளை அல்லது பல பொருள்களை விவரிப்பதற்கு வகுப்பு பயன்படுத்தப்படுகின்றது

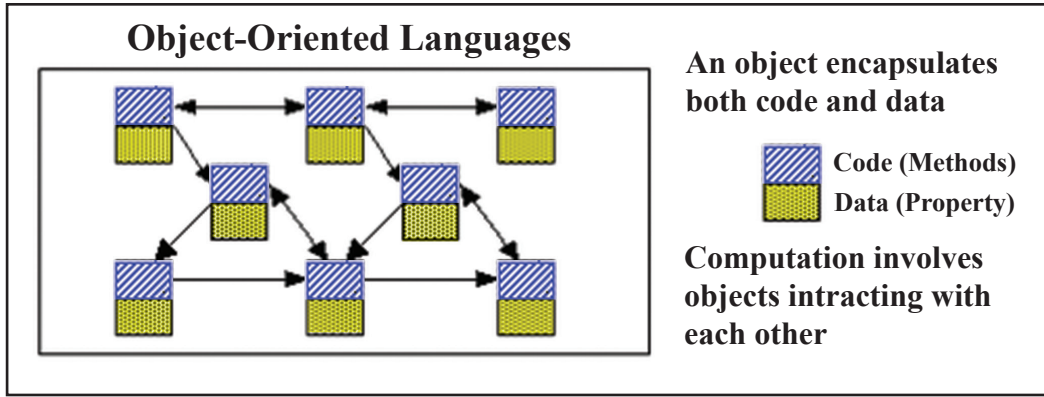


உரு 1.14





உரு 1.15



### செயற்பாடு



Car என்னும் வகுப்பைக் (Class) கருதும்போது அதன் பொருள்கள் (Objects), பண்புகள் (Properties) முறையியல்கள் (Methods) ஆகிய வற்றை இனங்காண்க.

### செய்நிரலாக்கமும் (Programming) பிரதி ஆவணமும் (Scripting)

செய்நிரல் மொழிகளின் சாதாரண பதத் தொடரியல் தொடர்பாகக் கடும் விதிகள் உள்ளன. நிதமும் தொகுத்தல் தேவை. செய்நிரல் மொழியைத் தொகுத்தல் (Syntax) வேண்டும். பிரதி ஆவண மொழிகள் பொதுவாக வேறு பிரயோகத்துடன் (உ-ம். வலை மேலோடி அல்லது வலைச் சேவையகம்) பெரும்பாலும் நடைமுறைப்படுத்தப்படும் ஒரு குறிமுறை உருவாக்கப்படுகின்றது. அவ்வாறே பிரதி ஆவண மொழியை விளக்க (Interpret) வேண்டும். புதியவர்களிடமிருந்து பிரதிச்சொல் மொழியைப் பயன்படுத்தல் மிக எளிதானது.

**Javascript, PHP** ஆகிய இரண்டும் பிரதி ஆவண மொழிகளாகும்.

### செயற்பாடு



பின்வரும் செய்நிரற் கட்டளைப்படிவங்களின் வேறுபாடுகளை ஒப்பிடுக.

- நடைமுறையும் வெளியீடும் (Procedural vs Declarative)
- கட்டமைப்பும் பொருள் சார்ந்ததும் (Structured vs Object oriented)
- செய்நிரலாக்கமும் பிரதி ஆவணமும் (Programming vs Scripting)

### 1.11.5 மொழிபெயர்ப்பு நிகழ்ச்சித் திட்டங்கள்

இயந்திர மொழி தவிர்ந்த வேறெந்த மொழியிலும் எழுதப்பட்ட செய்நிரல்களை (இலக்குப் பொருள் குறிமுறைகள்) இயக்கு முன்னர் இயந்திர மொழி அறிவுறுத்தல்களாக (இயந்திரக் குறிமுறை) மாற்ற வேண்டும்.

ஒருங்குசேர்ப்பு மொழியில் எழுதப்பட்ட ஒரு செய்நிரலை ஒருங்கு சேர்ப்பி எனப்படும் மொழிபெயர்ப்பு நிகழ்ச்சித்திட்டத்தின் மூலம் இயந்திர மொழி அறிவுறுத்தலாக மாற்ற வேண்டும்.

உயர்மட்டக் கணினி மொழியைப் பயன்படுத்தி எழுதப்பட்ட ஒரு செய்நிரலை இயந்திர மொழி அறிவுறுத்தல்களாக மாற்றுவதற்கு இரு நிகழ்ச்சித்திட்டங்கள் (மென்பொருள்கள்) பயன்படுத்தப்படுகின்றன.

1. வரிமொழிமாற்றி (Interpreter)
2. தொகுப்பி (Compiler)

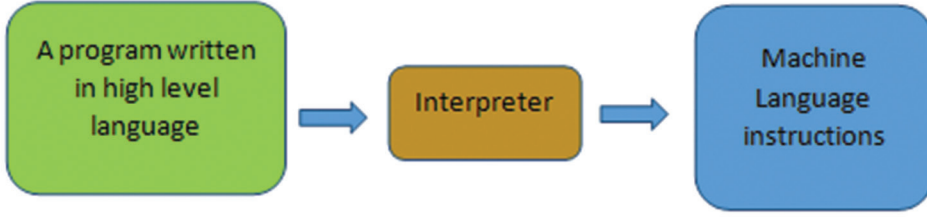
#### வரிமொழிமாற்றி

உயர் மட்டக் கணினி மொழியில் எழுதப்பட்ட செய்நிரலைச் செய்முறைப்படுத்து முன்னர் அதில் உள்ள கூற்றுகள் ஒவ்வொன்றையும் இயந்திர மொழிக் கட்டளையாக மொழிபெயர்த்து அக்கணத்தில் அதனை ஒத்த கட்டளையை இயக்கி மொழிபெயர்ப்பு முறைவழியாக்கம் நடைபெறும் மொழிபெயர்ப்பாளர் வரிமொழிமாற்றி எனப்படும்.

வரிமொழிமாற்றி பயன்படுத்தப்படும் கணினி மொழிகளில் செய்நிரலை மாற்றும்போது

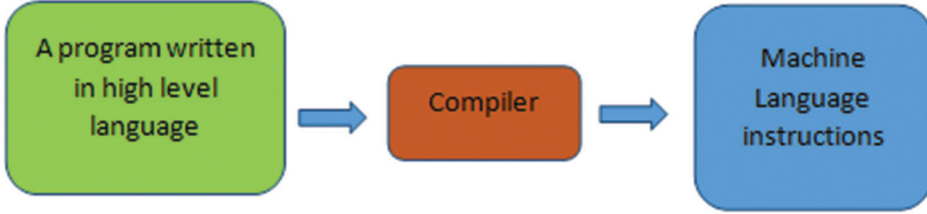
1. செய்நிரல் விதி வழக்கள் இல்லாவிடின் இயக்கிய பின்னர் சரியான வெளியீட்டைத் தருகின்றது.
2. செய்நிரலில் வழக்கள் இருப்பின் அதனை இறுதிவரைக்கும் செய்முறைப்படுத்த முடியாது (வழு உள்ள இடம் வரைக்கும் செய்முறைப்படுத்துவதற்கான ஆற்றல் உள்ளது).

குறிப்பு - செய்நிரல் இயக்கப்படும் ஒவ்வொரு தடவையும் மொழிபெயர்ப்பு நடைபெறும்.



### தொகுப்பி

உயர் மட்டக் கணினி மொழியில் எழுதப்பட்ட செய்நிரலைச் செய்முறைப்படுத்து முன்னர் அதனை முற்றாக ஒருதடவை இயந்திர மொழி அறிவுறுத்தல்களாக மாற்றிய பின்னர் இயக்குதல் தொகுப்பியின் மூலம் நடைபெறும்.

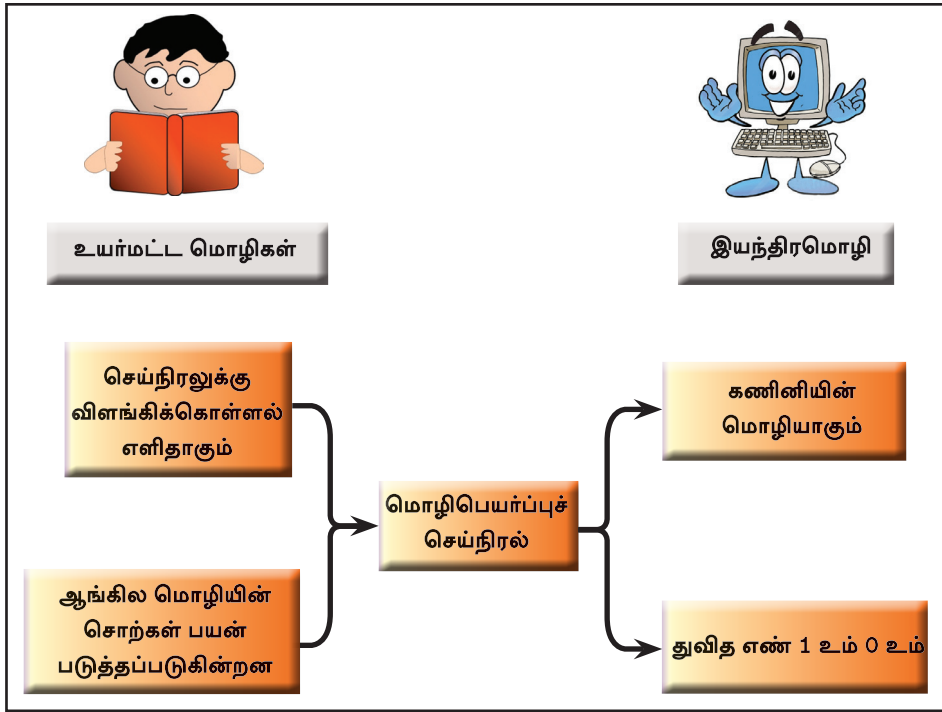


தொகுப்பி பயன்படுத்தப்படும் கணினி மொழியின் ஆதாரமூலக் குறிமுறையை (Source code) இயந்திரக் குறிமுறையாக மாற்றும்போது

1. செய்நிரலில் விதி வழுக்கள் இல்லாவிடின் இயக்கிய பின்னர் வெளியீடு வழங்கப்படும்.
2. செய்நிரலில் விதி வழுக்கள் இருப்பின் அதனை இயந்திரக் குறிமுறையாக மொழிபெயர்க்க முடியாது. அவ்வழு உள்ள இடங்கள் காட்டப்படும்.

**குறிப்பு** - செய்நிரலை ஒரு தடவை இயந்திரக் குறிமுறையாக மாற்றிய பின்னர் தேவையான எத்தனை தடவையும் செய்நிரலை இயக்கலாம். ஆதாரமூலக் குறிமுறை மாற்றப்பட்டால் மாத்திரம் மீண்டும் மொழிபெயர்ப்பு தேவைப்படும்.





உரு 1.17

## பொழிப்பு

- ஒரு பிரச்சினையைப் பகுப்பாய்வதன் மூலம் உள்ளீடு, முறைவழியாக்கம், வெளியீடு ஆகியவற்றை இனங்காணலாம்.
- நெறிமுறையை உருவாக்குவதற்குக் கோட்டுப் படங்களையும் போலிக்குறி முறையையும் பயன்படுத்தலாம்.
- தொடரி, தெரிவு, மீள்செயல் என்றவாறு கட்டுப்பாட்டுக் கட்டமைப்புகள் இருக்கின்றன.
- ஒரு நெறிமுறையில் இருக்கும் சில படிமுறைகளை அல்லது எல்லாப் படிமுறைகளையும் ஓர் ஆரம்பப் படிமுறையிலிருந்து இறுதிப் படிமுறை வரைக்கும் முறையே நடைமுறைப்படுத்தல் தொடரி (Sequence) எனப்படும்.
- ஒரு நிபந்தனையைத் திருப்திபடுத்தலுக்கு அல்லது திருப்திப்படுத்தாமைக்கு ஏற்ப நடைமுறைப்படுத்த வேண்டிய படிமுறை யாதெனத் தீர்மானிக்கும் சந்தர்ப்பம் தெரிவு (Selection) எனப்படும்.
- ஒரு நெறிமுறையில் இடம்பெறும் ஒரு படிமுறை அல்லது சில படிமுறைகள் அல்லது ஒரு நிபந்தனை திருப்தியாக்கப்படும் வரைக்கும் அல்லது திருப்தியாக இருக்கும் வரைக்கும் அல்லது திரும்பத் திரும்ப நடைபெறுதல் மீள்செயல் (Repetition) எனப்படும்.
- ஒரு மாறியை அல்லது மாறிலியை அல்லது செய்நிரலை அறிமுகஞ் செய்யப் பயன்படுத்தப்படும் பெயர் அடையாளங்காணி ஆகும்.
- செய்நிரலை நடைமுறைப்படுத்தும்போது அறிமுகஞ்செய்வதற்காக ஒதுக்கப் பட்ட பெறுமானங்கள் மாறுதலை அறிமுகஞ் செய்வதற்கு மாறி என்னும் பெயர் பயன்படுத்தப்படும்.
- பஸ்கால் என்பது ஓர் உயர்மட்டச் செய்நிரல் மொழியாகும்.
- ஒரு கோவையைச் சுருக்கல் செயல்களின் முன்னுரிமை ஒழுங்குமுறைக்கேற்ப நடைபெறுகின்றது.
- யந்திர மொழியும் ஒருங்குசேர்ப்பு மொழியும் கீழ் மட்டச் செய்நிரல் மொழிகளாகக் கருதப்படுகின்றன.
- பஸ்கால், பேசிக், சீ, ஜாவா என்னும் மொழிகள் உயர் மட்டச் செய்நிரல் மொழிகளுக்கு உதாரணங்களாகும்.
- யந்திரமொழியில் எழுதப்பட்ட செய்நிரலை நேரடியாக முறைவழியாக்கத் திற்குச் செலுத்தலாம்.
- யந்திரமொழி தவிர்ந்த வேறெந்தக் கணினி மொழியிலும் எழுதப்பட்ட செய்நிரலைச் செலுத்து முன்பாக யந்திரமொழி அறிவுறுத்தல்களாக மாற்றுதல் வேண்டும்.
- வரிமொழிமாற்றி (Interpreter), தொகுப்பி (Compiler) ஆகியன இரு மொழி பெயர்ப்புச் செய்நிரல்களாகும்.