තුන්වන ලිපිය

ක්ෂුද්‍ර පාලන ඒකක ක්‍රමලේඛනය කිරීම සඳහා අවශ්‍ය වන මෘදුකාංග

ආරම්භයේ දී ම මෙම පාඩම් මාලාව සඳහා දෘඩාංග (Hardware) හා මෘදුකාංග (Software) යන දෙවර්ගය ම භාවිත කිරීමට සිදු වන බව අප සඳහන් කළා ඔබට මතක ඇතැ යි සිතමු. අද අපි මයික්‍රොකොන්ට්‍රෝලර් ක්‍රමලේඛනය සඳහා අවශ්‍ය වන මූලික මෘදුකාංග පිළිබඳව හදාරමු. මේ සඳහා ඔබට මෘදුකාංග පිළිබඳ විශේෂ දැනුමක් අත්‍යවශ්‍ය නො වන බව සලකන්න. හැකිතාක් සරල ව කරුණු ඉදිරිපත් කිරීමට අපි උත්සුක වෙමු.

මුලින් ම μC හා පරිගණක ක්‍රමලේඛනය කිරීම සඳහා යොදා ගන්නා භාෂා (Programming languages) ගැන යම් අවබෝධයක් ලබා ගනිමු. පහත උදාහරණය සලකන්න.

අපි එදිනෙදා ජීවිතයේ දී අදහස් හුවමාරු කිරීමට භාෂාවක් භාවිත කරමු. මෙය සිංහල, දෙමළ, ඉංග්‍රීසි ආදී කවර හෝ භාෂාවක් විය හැකි ය. එහෙත් මෙහි දී අප භාවිත කරන භාෂාව අනික් පාර්ශ්වයට තේරුම් ගත හැකි එකක් විය යුතු ය. එසේ නො වුව හොත් නිවැරැදි අදහස් හුවමාරුවක් සිදු නො වේ. එවන් අවස්ථාවල දී අපට පරිවර්තකයකු ගේ සේවය අවශ්‍ය වනු ඇත.

මේ ආකාරයට ම මයික්‍රොන්ට්‍රෝලරවලට ද (එසේ ම පරිගණකවලට ද) වැටහෙන භාෂාවක් පවතී. මේ භාෂාව අපි යන්ත්‍ර භාෂාව ලෙස හඳුන්වමු. තාක්ෂණික ව්‍යවහාරයේ දී Machine language ලෙස හැඳින්වෙන මෙම යන්ත්‍ර භාෂාවෙහි ඇත්තේ සංකේත දෙකක් පමණි. ඒවා නම් "1" සහ "0" යන ඉලක්කම් දෙක පමණි. සරල සංඛ්‍යාංක (Digital) උපකරණවල සිට සංකීර්ණ පරිගණක දක්වා වූ ඉලෙක්ට්‍රෝනික උපකරණවල ක්‍රියාකාරීත්වය හැසිරවීමට මෙම සංකේත දෙක ප්‍රමාණවත් වේ. "1" සහ "0"හි විවිධ සංයෝජනයන් භාවිත කර ඉහත සඳහන් උපකරණවලට විවිධ පණිවිඩ, විධාන, අගයයන් ලබා දිය හැකි ය. එමෙන් ම ඒවායින් බාහිර ලෝකයට (For users) තොරතුරු ලබා දෙන්නේත් "1" සහ "0" යන සංඛ්‍යාවලින් සමන්විත සංයෝජනයන් ලෙසිනි.

පහත උදාහරණය ද සලකා බලන්න.

"1" සහ "0" සංඛ්‍යා අටක් භාවිත කොට ඔබට වෙනස් සංයෝජන 256ක් ලබා ගත හැකි ය. (0 සිට 255 දක්වා) මෙම ඕනෑ ම සංයෝජනයක් 00000000 සිට 11111111 දක්වා වූ පරාසය තුළ පවතී. එනම් ඔබට එකිනෙකට වෙනස් විධාන හෝ පණිවුඩ 256ක් නිර්මාණය කරගත හැකි බව ය.

තව ද සිංහල භාෂාවේ අක්ෂර 56ක් ද, ඉංග්‍රීසි භාෂාවේ අක්ෂර 26ක් ද ඇත. එම අක්ෂරවල විවිධ සංයෝජන මගින් විවිධ අරුත් ඇති වචන, වාක්‍ය තැනිය හැකි අතර ඒවා අපට තේරුම් ගත හැකි වේ. එහෙත් පරිගණකයකට, මයික්‍රොකොන්ට්‍රෝලරයකට හෝ වෙනයම් සංඛ්‍යාංක (Digital) උපකරණයකට ඒවා තේරුම් ගත නොහැකි ය. මන්ද යත් එම උපකරණ තේරුම් ගන්නා එක ම භාෂාව යන්ත්‍ර භාෂාව ("1" සහ "0") වීම ය.

එසේ නම් අපේ අවශ්‍යතාවන්, විධානයන්, පරිගණකයකට හෝ මයික්‍රොකොන්ට්‍රෝලරයකට පැවසීමට ඒවා "1" සහ "0"හි සංයෝජනයන් බවට පත් කළ යුතු ය. එනම් යන්ත්‍ර භාෂාවට හැරවිය යුතු ය.

මෙය කළ හැකි ද?

සත්‍ය වශයෙන් ම පරිගණක ලෝකයේ ආරම්භයේ දී සිටි නිර්මාපකයින්ට මෙය සිදු කළ යුතු විය. එහෙත් මෙය ඉතා කරදරකාරී, කාලය වැය වන ක්‍රියාවලියකි. එසේ ම මෙහි දී සිදු වන වැරැදීමක් නිවැරැදි කිරිම හෝ අඩු තරමින් වරද සොයා ගැනීම දුෂ්කර කටයුත්තක් බව ඔබට වැටහෙනු ඇත. එසේ ම යන්ත්‍ර භාෂාවෙන් ලියූ ක්‍රමලේඛයක් කියවා තේරුම් ගැනීම ද අති දුෂ්කර කටයුත්තකි.

ඊට හේතුව අප හුරු වී ඇත්තේ අප ගේ මවු භාෂාවට (හෝ තවත් ස්වාභාවික භාෂා කිහිපයකට) පමණක් වීම ය. එම නිසා අපි අප ගේ භාෂාව භාවිත කිරීමට කැමැත්තෙමු. එහෙත් මයික්‍රොකොන්ට්‍රෝලරයට එය නො තේරෙන බැවින් යම් ආකාරයක පරිවර්තකයකු ගේ සහාය අවශ්‍ය වේ. මෙම පරිවර්තකයන් තාක්ෂණික ව්‍යවහාරයේ දී Compilers ලෙස හැඳින්වේ. මේවායේ කාර්යයන් පරිවර්තනයට පමණක් සීමා නො වේ. එමගින් තවත් බොහෝ කාර්යයන් ඉටු කරයි. ඒ ගැන සාකච්ඡා කිරීම පසුවට කල් තබමු.

Compiler එකක් යනු ඔබේ පරිගණකයේ පිහිටුවා ගත හැකි මෘදුකාංගයක් වේ. (An application program which can be installed in your computer) මෙමගින් අප ලියන ක්‍රමලේඛනයන් යන්ත්‍ර භාෂාවට හරවා පරිගණකයකට හෝ මයික්‍රොකොන්ට්‍රෝලරයකට තේරුම් ගත හැකි පරිදි සකස් කෙරේ.

අපට තේරුම් ගැනීමට පහසු වන පරිදි ක්‍රමලේඛනයන් රචනා කිරීමට බොහෝ විට ඉහළ මට්ටමේ ක්‍රමලේඛන භාෂාවන් (High-level programming languages) යොදා ගැනේ. High - level programming languages යනු සාමාන්‍යයෙන් යම් නිශ්චිත වූ සංකේතයන් හා විධානයන් සමූහයකින් සැදුම් ලත් ඉංග්‍රීසි භාෂාවට ආසන්න වශයෙන් සමාන වූ භාෂාවන් වේ. උදාහරණ වශයෙන් 'C, C++, Java, VB' ආදිය හඳුන්වා දිය හැකි ය. මෙම ක්‍රමලේඛන භාෂාවන් ගෙන් රචිත ක්‍රම ලේඛනයන් පහසුවෙන් වටහා ගත හැකි වේ. යම් පරිගණක භාෂාවක් (Programming language) High - level language ලෙස හඳුන්වයි නම්, එය බොහෝ විට මිනිසුන්ට කියවා තේරුම් ගත හැකි අක්ෂර, සංකේත, විධාන ආදියෙන් යුක්ත වේ. මේවා බොහෝ විට ඉංග්‍රීසි භාෂාවට සමාන වන නිසා, ඉංග්‍රීසි භාෂාව පිළිබඳ මූලික දැනීමක් ඔබට තිබිය යුතු බව අපි මුල දී සඳහන් කළෙමු.

අංක 1 රූප සටහනින් විවිධ ක්‍රමලේඛන භාෂාවන් හා ඒවා කෙතරම් අපට සමීප දැ යි යන්න දැක්වේ.

ස්වාභාවික භාෂා (සිංහල, ඉංග්‍රීසි, දෙමළ ආදී)

↑

'C', 'VB' වැනි ඉංග්‍රීසියට සමීප
High - level languages

Machine language එකට වඩාත් සමීප
Assembly language

1 සහ 0 පමණක් යෙදෙන
Machine language

(ඉතිරි කොටස ලබන සතියේ)

මොරටුව විශ්ව විද්‍යාලයයේ විද්‍යුත් හා විදුලි සංදේශ අංශයේ
ගාමිණී ජයසිංහ/කෝලිත ධර්මප්‍රිය

අනුමාන තර්ක (Fuzzy Logic)

සිව්වන ලිපිය

සාමාන්‍ය කුලක පද්ධතියකට අනුමාන තර්ක ක්‍රමය යෙදිය හැක්කේ කෙසේ දැ යි බලමු. කුලක (Sets) සහ ඒ සම්බන්ධ නීති (Set Theory) අනුමාන කුලක (Fuzzy sets) සඳහා යෙදෙන්නේ කෙසේ දැ යි ඔබට කුතුහලයක් ඇති වනු නොඅනුමාන ය. පසුගිය පාඩම්වල ගත් වර්ගීකරණය (Classification) උදාහරණයේ දී අනුමාන තර්ක ක්‍රමය භාවිත නො කොට එනම් සාමාන්‍යයෙන් භාවිත කරන විට එක පුද්ගලයකු (කුලක ආශ්‍රිත ව සලකන විට එක් අවයවයක්) එක් වර්ගීකරණයකට (එක් කුලකයකට) පමණක් අයත් වේ. එසේ ම සැම පුද්ගලයකු ම එක් කුලකයකට හෝ අයත් වේ. මෙහි දී කුලක ඡේදනයක් ඇති නො වේ.

අද අපි මෙයට වඩා වෙනස් උදාහරණයක් සලකමු. එනම් සාමාන්‍යයෙන් භාවිත කරන කුලකවල එක් අවයවයක් එක ම කුලකයකට වඩා අයත් විය හැකි, එලෙස ම සමහර අවයව කිසි ම කුලකයකට අයත් නො වන (සර්වත්‍ර කුලකයට පමණක්) අවස්ථාවක් සලකමු. පන්තියක සිසුන් 40ක් ඇති අතර මෙයින් 12 දෙනකු ගණිතයට දක්ෂ අතර (A) 20ක් සංගීතයට දක්ෂ ද (B), 20 ම දක්ෂ සිසුන් 6 දෙනකු (A∩B) සිටින බව සලකමු.

එවිට ගණිතයට පමණක් දක්ෂ සිසුන් 6ක් ද, සංගීතයට පමණක් දක්ෂ සිසුන් 14ක් ද, දෙකට ම අදක්ෂ සිසුන් 14ක් පමණක් ලැබේ. අනුමාන තර්ක ක්‍රමය යටතේ මෙය වඩාත් සංකීර්ණ වේ. අද පාඩමේ අරමුණ, අනුමාන තර්ක ක්‍රමය යටතේ අවයව ගණන සෙවීම නො ව, සාමාන්‍ය තර්ක ක්‍රමය යටතේ ඉතා ම සරල, එහෙත් අනුමාන තර්ක ක්‍රමය යටතේ මඳක් සංකීර්ණ අදහසක් විමසීම ය. යම් සිසුවකු සලකමු. ඔහු ගණිතයට පමණක් දක්ෂ නම්, ඔහු පහත කුලකවලට අයත් වීමේ සත්‍යතාව සාමාන්‍ය තර්ක ක්‍රමය යටතේ මෙසේ දැක්විය හැකි ය.

$A\ (x) = 1$
$B\ (x) = 0$
$A \cap B(x) = 0$
$A \cup B(x) = 1$

සංගීතයට පමණක් දක්ෂ සිසුවකු නම්

$A\ (x) = 0$
$B\ (x) = 1$
$A \cap B(x) = 0$
$A \cup B(x) = 1$

දෙකට ම අදක්ෂ සිසුවකු නම්

$A\ (x) = 1$
$B\ (x) = 1$
$A \cap B\ (x) = 1$
$A \cup B\ (x) = 1$

මෙහි දී අනුමාන තර්ක ක්‍රමය යටතේ යම් සිසුවකු යම් කුලකයකට අයිති වීමේ සත්‍යතාව සැම විට ම "0" හෝ "1" නොගන්නා අතර, ඒ අතර ඇති ඕනෑ ම අගයක් ගත හැකි ය. මෙහි දී ඉහත ආකාරයෙන් පහසුවෙන් දැක්විය නොහැකි ය. මේ සඳහා ක්‍රමයක් මහාචාර්යය සදේ ඉදිරිපත් කර ඇත. ඔහු ලොවෙහි ප්‍රථම වතාවට අනුමාන කුලක (Fuzzy sets) පිළිබඳව ලොවට හඳුන්වා දුන් පුද්ගලයා වේ. ඔහු ගේ මෙම ලිපිය 1965 දී පළ වුණ නමුත් එය විකාර අදහසක් ලෙස නොයෙකුත් විවේචන ද එල්ල විය. පසුව 80 දශකයේ කෘත්‍රිම බුද්ධිය පිළිබඳ පර්යේෂණයන් සඳහා මෙම සංකල්පය ජනප්‍රිය වී ගිය අතර ඔහු අනුමාන කුලක පිළිබඳ පියා (Father of Fuzzy sets) ලෙස ද හඳුන්වනු ලබයි.

මෙම උදාහරණයට අනුව සිසුන් 3ක ගේ (x, y සහ z) ගණිතයට දක්ෂ වීම සහ සංගීතයට දක්ෂ වීම 4.1 වගුව ආශ්‍රයෙන් දැක්විය හැකි නම්

සිසුවා	A කුලකයේ අවයවයක් වීමේ සත්‍යතාව F(A)	B කුලකයේ අවයවයක් වීමේ සත්‍යතාව F(B)
x	0.9	0.3
y	0.8	0.7
z	0.6	1.0

4.1 වගුව

මෙහි දී අනුමාන සත්‍යතාව (Fuzzy membership value) F යන සංකේතයෙන් හඳුන්වා දී ඇත. මේ අනුව සෛද්ධාන්තික වශයෙන් (මහාචාර්යය සදේට අනුව)

$$F_{(A \cap B)} = \text{අවම}\left(F_A, F_B\right) \rightarrow (4.1)$$

$$F_{(A \cup B)} = \text{උපරිම}\left(F_A, F_B\right) \rightarrow (4.2)$$

$$F(A)' = 1 - F(A) \rightarrow (4.3)$$

ප්‍රකාශන දැක්විය හැකි ය.

එම ප්‍රකාශන වඩාත් පැහැදිලි කිරීම සඳහා x සිසුවා උදාහරණයක් ලෙස සලකමු. 4.1 වගුවට අනුව x ගණිතයට දක්ෂතාව $F\ (A)_x = 0.9$ වේ. මෙහි අදහස x සිසුවා ගණිතයට දක්ෂ යැ යි යමකු පැවසුව හොත් එහි සත්‍යතාව 0.9ක් බවයි. එවිට එම සිසුවා ගණිතයට දක්ෂ නොවීමේ සත්‍යතාව $\left(F\ (A)'_x\right)$ 0.1 විය යුතු ය. මෙම අදහස 4.3 ප්‍රකාශනය මගින් දැක්වේ. එලෙස ම එම සිසුවා සඳහා සංගීතයට ඇති දක්ෂතාවේ සත්‍යතාව මෙසේ පැහැදිලි කළ හැකි ය.

සංගීතයට දක්ෂ වීමේ සත්‍යතාව $F\ (B)\ x = 0.3$

සංගීතයට දක්ෂ නො වීමේ සත්‍යතාව $F\ (B)'x = 0.7\ (1\text{-}0.3)$

දැන් මෙම සිසුවා විෂයයන් 2කට ම සත්‍ය යැ යි කිසිවකු පැවසුව හොත් එහි සත්‍යතාව ගැන ඔබට කිව හැක්කේ කුමක් ද? මෙහි දී ඔහු ගණිතයට දක්ෂ බව වඩාත් සත්‍යතාවට ආසන්න වේ. (0.9). සංගීතයට දක්ෂ බව එම අගයට වඩා අඩු වේ. (0.3). දෙකට ම පොදු සත්‍ය බව

වැනි පිටුවට

මොරටුව විශ්වවිද්‍යාලයයේ යාන්ත්‍රික ඉංජිනේරු අංශයේ
ජ්‍යෙෂ්ඨ කථිකාචාර්ය
ආචාර්ය පාලිත දසනායක