સમસ્યા અને સમસ્યાનું નિરાકરણ

દરેક મનુષ્ય પોતાની રોજિંદી જિંદગીમાં અનેક સમસ્યાઓનું નિરાકરણ લાવે છે. ચાલો, આપણે આવી એક ક્રિયાની ચર્ચા કરીએ. ધારો કે તમે તમારા મિત્રો સાથે સંતાકુકડી રમત (Hide and seek) રમો છો. આ રમતમાં આપણી સમસ્યા એ છે કે આપણા છુપાયેલા મિત્રની જગ્યા શોધવાની છે. આ કાર્ય ફક્ત તો જ થઈ શકે જો આપણે તેની છુપાવાની જગ્યા વિશે આગાહી કરી શકીએ.

ચાલો, આપણે બીજી એક સમસ્યા લઈએ, ધારો કે આપણે એક ગાણિતિક સમીકરણનો ઉકેલ લાવવો છે જેમાં આપણને $2x + 4 = 0$ સમીકરણ આપેલું છે તો xની શું કિંમત હોઇ શકે ? આ સમસ્યાના જવાબ માટે આપણે સમીકરણને $2x = -4$ની રીતે ગોઠવણી કરવી પડશે, અને પછી $x = -4/2$, આથી અહીં જવાબ -2 મળશે.

ચાલો, આપણે અન્ય ઉદાહરણ લઈએ. આપણે અંગ્રેજી શબ્દ "Eloquent" નો અર્થ શોધવો છે. આ માટે આપણે એક અંગ્રેજી ભાષાનો શબ્દકોશ જોઈશે. શબ્દકોશમાં હજારો શબ્દો અને તેના અર્થ હોય છે. છતાં આપણે ઘણી ઝડપથી ઈચ્છિત શબ્દનો અર્થ શોધી કાઢીએ છીએ. અત્યાર સુધીમાં ક્યારેય પણ તમે વિચાર્યું છે કે આ કાર્ય તમે આટલી ઝડપથી કઈ રીતે કરી શકો છો? હકીકતમાં શબ્દકોશમાં શબ્દો ક્રમિક રીતે ગોઠવાયેલા હોય છે તેનો આપણે ફાયદો લઈએ છીએ. આપણે જે શબ્દો "E" થી શરૂ ન થતા હોય તેને ઝડપથી કાઢી નાખીએ છીએ અને "E" થી શરૂ થાય તેવા શબ્દો શોધવાનું શરૂ કરીએ છીએ, એ જ રીતે બીજો અક્ષર "l" હોય તેવા શબ્દો શોધવા માટે કાઢી નાખવાની પદ્ધતિ (elimination method) નો ઉપયોગ કરીએ છીએ. આપણે આ કાઢી નાખવાની પદ્ધતિનો જ ઉપયોગ બાકીના અક્ષરો શોધવામાં ચાલુ રાખીએ છીએ કે જ્યાં સુધી આપણો ઈચ્છિત શબ્દ મળી ન જાય.

ઉપરના ઉદાહરણથી આપણને ખ્યાલ આવ્યો કે સમસ્યા કેવી હોઈ શકે ? સમસ્યાઓનું નિરાકરણ માનસિક શક્તિઓ અથવા યંત્રોના કાર્ય (કોઈ પદ્ધતિસરના કાર્ય) વપરાશકર્તાનું ઈચ્છિત પરિણામ મેળવવા તરફ લઈ જાય છે. ઉપર ચર્ચા કરેલાં ઉદાહરણો દર્શાવે છે કે કેટલીક સમસ્યાઓના પરિણામની આગાહી ચોક્કસ રીતે કરી શકાય છે જ્યારે કેટલીક સમસ્યાઓના ચોક્કસ પરિણામની આગાહી કરવી ઘણી મુશ્કેલ છે. આ રીતે સમસ્યાઓનું બે પ્રકારમાં વર્ગીકરણ કરી શકાય : સારી રીતે વ્યાખ્યાયિત કરેલી અને સ્પષ્ટ રૂપરેખા વગરની સમસ્યાઓ. ઉપર જણાવેલાં ઉદાહરણોમાં બીજો અને ત્રીજો દાખલો એ સ્પષ્ટ રીતે વર્ણવેલી સમસ્યાનો છે જ્યારે પહેલો દાખલો એ સ્પષ્ટ રૂપરેખા વગરનો છે. અહીં તમે નોંધ કરશો કે સ્પષ્ટ સમસ્યાઓનાં લક્ષ્ય ચોક્કસ અને સ્પષ્ટ હોય છે અને આથી આપણે સમસ્યાના ઉકેલનાં તબક્કા (પગલાં) પણ સ્પષ્ટ રીતે વર્ણવી શકીએ છીએ. એ પણ હકીકત છે કે કમ્પ્યૂટર ફક્ત સ્પષ્ટ રૂપરેખા ધરાવતી સમસ્યાઓનો જ ઉકેલ લાવે છે અને આથી આપણે આ પ્રકરણમાં ફક્ત સ્પષ્ટ રૂપરેખા ધરાવતી (well defined) સમસ્યાઓ બાબત જ ચર્ચા કરીશું.

કમ્પ્યૂટરનાં ક્ષેત્રમાં કોઈ પણ આપેલી સમસ્યાઓનો ઉકેલ એ કમ્પ્યૂટરને આપેલી ક્રમિક સૂચનાઓ જ છે. કમ્પ્યૂટર અતિ સરળતાથી ઘણી ગૂંચવાડાભરી વિવિધ પ્રકારની સમસ્યાઓને ઉકેલે છે. સમસ્યાના ઉકેલ માટે કમ્પ્યૂટરને સૂચનાઓનો સંપૂર્ણ ગણ આપવો પડે છે. આ સૂચનાઓ કમ્પ્યૂટરને દરેક તબક્કે શું કરવાનું છે તે જણાવે છે. એક વસ્તુ ખાસ યાદ રાખો કે કમ્પ્યૂટર જાતે સમસ્યાનો ઉકેલ લાવી શકતું નથી પણ તે સમસ્યાના ઉકેલમાં ફક્ત મદદરૂપ થાય છે. નીચે જણાવેલાં પગલાંને અનુસરવાથી આપણે કોઈ પણ સમસ્યાનો ઉકેલ લાવી શકીએ છીએ :

1. સમસ્યાને વ્યાખ્યાયિત કરો.
2. ઈનપુટ, આઉટપુટ અને સમસ્યાના અવરોધોને ઓળખો.
3. સમસ્યાના ઉકેલના અલગ અલગ વિકલ્પો શોધો.
4. ઉપર જણાવેલા વિવિધ વિકલ્પોની યાદીમાંથી સારામાં સારો વિકલ્પ પસંદ કરો.
5. આપણે નક્કી કરેલા વિકલ્પ માટે વિગતવાર ક્રમિક સૂચનાઓ તૈયાર કરો.
6. આપણે તૈયાર કરેલા સૂચનાઓના ગણ દ્વારા પરિણામની ગણતરી કરો.
7. આપણે મેળવેલો જવાબ સાચો છે કે કેમ તે ચકાસો.

જે વ્યક્તિએ ઉકેલ મેળવવો છે તેણે પગલાં 1થી 5 કરવાનાં છે જ્યારે પગલાં 6 અને 7 કમ્પ્યૂટર વડે થશે.

ધારો કે આપણે કોઈ આપેલી સંખ્યા એકી છે કે બેકી તે શોધવું છે. આ પ્રશ્નના ઉત્તર માટે નીચે આપેલી સૂચનાઓનો ઉપયોગ કરી શકાય :

1. સંખ્યાનું ઇનપુટ લો.
2. તે સંખ્યાને 2 વડે ભાગો અને શેષ શોધો.
3. જો શેષ 1 રહે તો આપેલી સંખ્યા એકી છે નહીંતર તે સંખ્યા બેકી છે.

આ સમસ્યાનો સર્વસામાન્ય ઉકેલ નીચે જણાવેલી ત્રણ રીતો વડે મેળવી શકાય :

1. સુડો કોડ (Pseudo code) - અનુકરણ કોડ
2. ફ્લોચાર્ટ (Flowchart) - ક્રમદર્શી રેખાચિત્ર
3. અલ્ગોરિધમ (Algorithm) - કલનવિધિ

આપેલી સંખ્યા એકી છે કે બેકી તે જાણવા માટે ઉપર જણાવેલાં ત્રણ પગલાં એ સુડો કોડ (Pseudo code) છે. સુડો એટલે આભાસી અથવા અનુકરણ કરેલો (simulated). આપણે આમાંનો બીજો અર્થ અહીં સ્વીકારીશું કારણ કે અહીં આપણા સંદર્ભમાં તે વધારે બંધબેસતો શબ્દ છે. આપણે આ પ્રશ્નના ઉકેલ માટે ઉપર જણાવેલાં ત્રણ પગલાંને અનુકરણ કરેલો કોડ કહી શકીએ. ચાલો, હવે અન્ય બે રીતો વિશે પણ જાણીએ.

ફ્લોચાર્ટ (Flowchart)

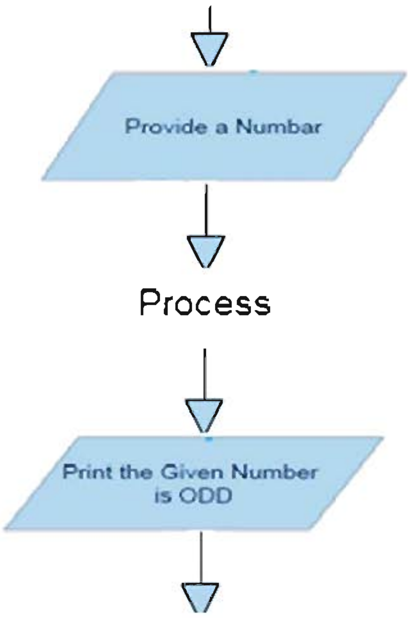
ફ્લોચાર્ટ એ એક કાર્યરીતિ છે જેમાં સમસ્યાના ઉકેલ માટે યંત્ર દ્વારા કરવામાં આવતા દરેક કાર્યને ચિત્રાત્મક સ્વરૂપે રજૂ કરવામાં આવે છે. જુદા જુદા કાર્યો માટે વિવિધ ચિહ્નોનો ઉપયોગ ફ્લોચાર્ટ બનાવવામાં થાય છે. આ ચિહ્નોને ફ્લોચાર્ટનાં ઘટકો કહેવામાં આવે છે. આપણી પાસે પ્રક્રિયાના દરેક કાર્ય માટે અલગ અલગ (અજોડ) ચિહ્ન છે. ચાલો, હવે આપણે વધારે વપરાતા ઘટકો અને તેનાં ચિહ્નો બાબત ચર્ચા કરીએ.

આરંભ અને અંત (Start and End) : આરંભ અને અંત ઘટકો ફ્લોચાર્ટની શરૂઆત અને અંત બતાવવા માટે વપરાય છે. તે આકૃતિ 9.1માં દર્શાવ્યા પ્રમાણે અંડાકાર હોય છે. આ ચિહ્નોને અંતિમ ચિહ્નો (ટર્મિનલ સિમ્બોલ) પણ કહેવામાં આવે છે.

આકૃતિ 9.1 : ટર્મિનલ સિમ્બૉલ

કોઈ પણ ફ્લોચાર્ટમાં આ ચિહ્ન બે વાર વપરાય છે : ફ્લોચાર્ટની શરૂઆતમાં તેમજ તેના અંતમાં.

ઇનપુટ - આઉટપુટ (Input-Output) : દરેક સમસ્યાને ઇનપુટ (નિવેશ)ની જરૂર હોય છે, આ નિવેશ ઉપર પ્રક્રિયા કરવામાં આવે છે અને તે આઉટપુટ (નીપજ-ઉત્પાદન) તૈયાર કરે છે. ફરી ઉપર જણાવેલો પ્રશ્ન 3 લો, અહીં આપણે એક સંખ્યાના ઇનપુટની જરૂર છે, આ આપેલી સંખ્યાને 2 વડે ભાગવાની અને શેષ શોધવાની એ પ્રક્રિયા છે અને આપણે આપેલી સંખ્યા એકી છે કે બેકી તે નિર્ણય એ આપણું આઉટપુટ છે. આથી આપણે એક ઇનપુટ અને એક આઉટપુટ દર્શાવવા માટે ચિહ્ન જોઈશે. ફ્લોચાર્ટમાં ઇનપુટ અને આઉટપુટ દર્શાવવા માટે આકૃતિ 9.2માં દર્શાવ્યા પ્રમાણે સમાંતર ચતુર્ભુજ વપરાય છે.

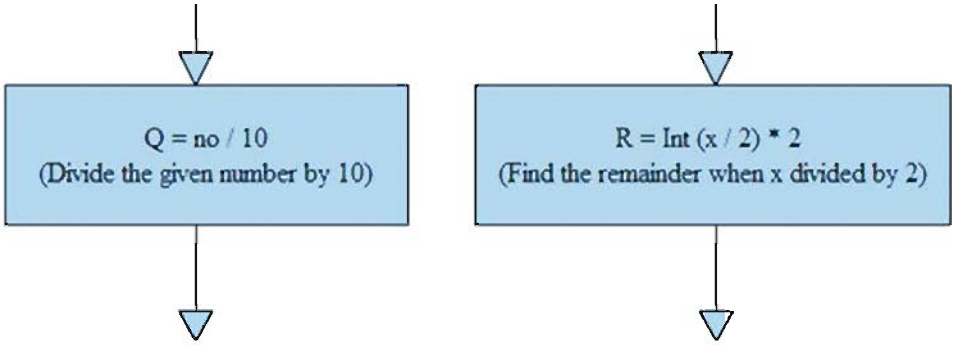


આકૃતિ 9.2 : ઈનપુટ-આઉટપુટ માટેનું ચિહ્ન

તીર (ઍરો - Arrow) : આકૃતિ 9.1 અને આકૃતિ 9.2નું નિરીક્ષણ કરો તો જણાશે કે આપણે ચિહ્નની અંદર જતું તેમજ ચિહ્નમાંથી બહાર આવતું તીર (ઍરો) બતાવેલું છે. કાર્ય જે ક્રમમાં કરવાનું હોય તે દર્શાવવા માટે તીરનો ઉપયોગ થાય છે. સામાન્ય રીતે તે ચિહ્નથી શરૂ થાય છે અને બીજા ચિહ્ન ઉપર તેનો અંત આવે છે. આ રીતે એક તીર એક ચિહ્નમાંથી નીકળે છે અને એક તીર તે ચિહ્નમાં અંદર જાય છે.

અહીં નોંધ કરશો કે start ચિહ્નમાંથી ફક્ત બહાર જતું તીર છે જ્યારે End ચિહ્નમાં ફક્ત તેની અંદર જતું તીર છે.

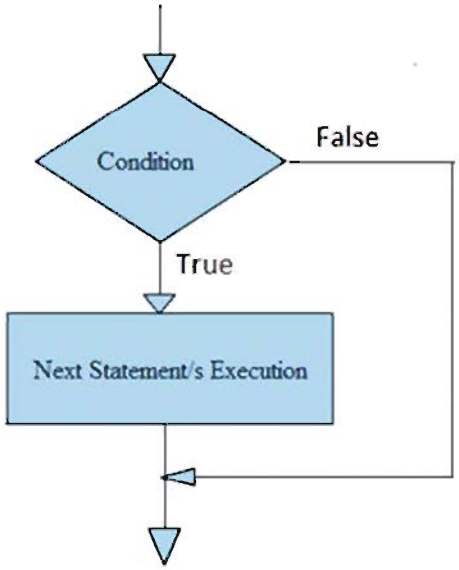
પ્રક્રિયા (Process) : કોઈ પણ સમસ્યાના ઉકેલમાં પ્રક્રિયા (process) એ મુખ્ય ભાગ છે. હકીકતમાં પ્રક્રિયા એ ક્રિયાઓની એક શ્રેણી છે. પ્રક્રિયાને દર્શાવવા માટે આકૃતિ 9.3માં દર્શાવ્યા પ્રમાણે એક લંબચોરસનું ચિહ્નનો આપણે ઉપયોગ કરીએ છીએ.



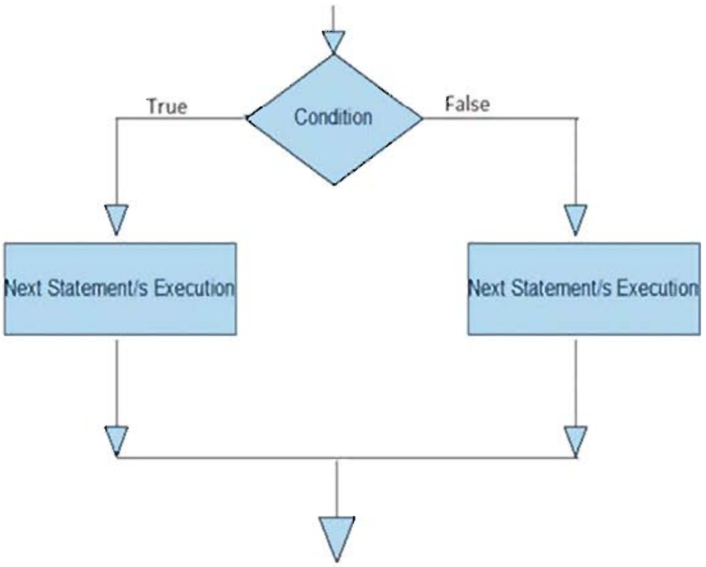
આકૃતિ 9.3 : પ્રક્રિયાનું ચિહ્ન

સામાન્ય રીતે કમ્પ્યૂટરની પ્રક્રિયા એ અંકગણિતનું કોઈ કાર્ય અથવા તાર્કિક ક્રિયા હોય છે. અંકગણિતનું કાર્ય એટલે સરવાળા, બાદબાકી, ગુણાકાર અથવા ભાગાકાર. તાર્કિક કાર્ય સામાન્ય રીતે નિર્ણય લેવામાં મદદરૂપ થાય છે જ્યારે તે કોઈ પ્રશ્નના હા કે ના પ્રકારના જવાબ માટે વપરાય છે. દાખલા તરીકે, આપણે પ્રશ્ન પુછીએ કે 10 એ 5 કરતાં મોટો છે ? આ પ્રશ્નનો જવાબ 'હા' અથવા સાચો છે.

નિર્ણય (Decision) : તાર્કિક નિર્ણયની પ્રક્રિયાને આકૃતિ 9.4માં દર્શાવ્યા પ્રમાણે એક હીરા આકારનાં ચિહ્નથી દર્શાવવામાં આવે છે. તેને પરીક્ષણ ચિહ્ન (ટેસ્ટ સિમ્બોલ) પણ કહેવામાં આવે છે. જ્યારે આપણે ઉકેલ માટેની સામાન્ય શ્રેણી બદલવા ઇચ્છતા હોય (જુઓ આકૃતિ 9.4) અથવા નિર્ણયનાં પરિણામનાં આધારે કોઈ ચોક્કસ વિધાનનો અમલ કરવાની જરૂર હોય (જુઓ આકૃતિ 9.5) ત્યારે નિર્ણયપેટી (decision box) વપરાય છે.

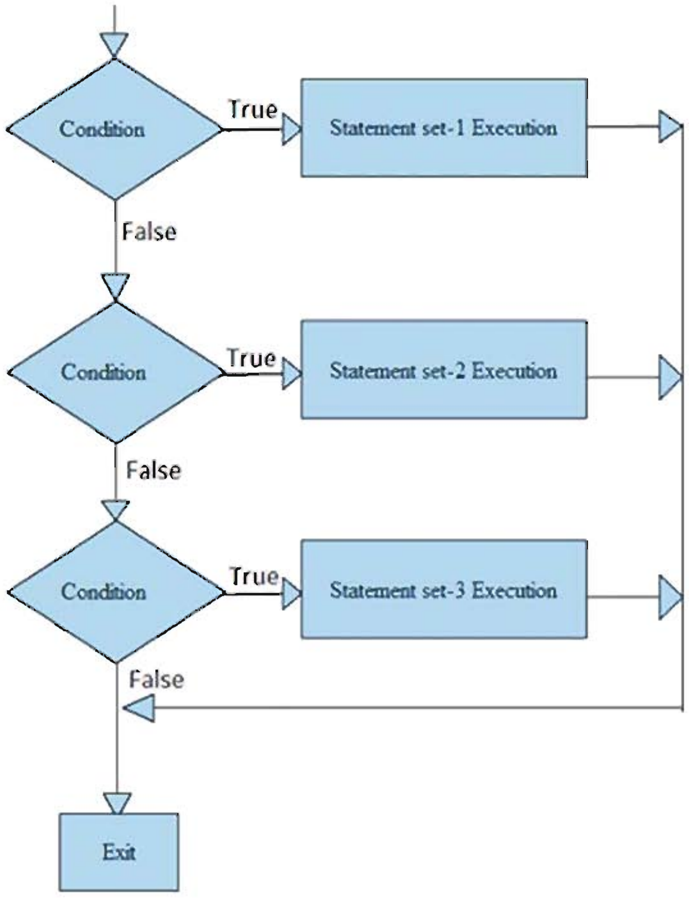


આકૃતિ 9.4 : નિર્ણય ચિહ્ન



આકૃતિ 9.5 : જુદાં જુદાં પરિણામોનો અમલ

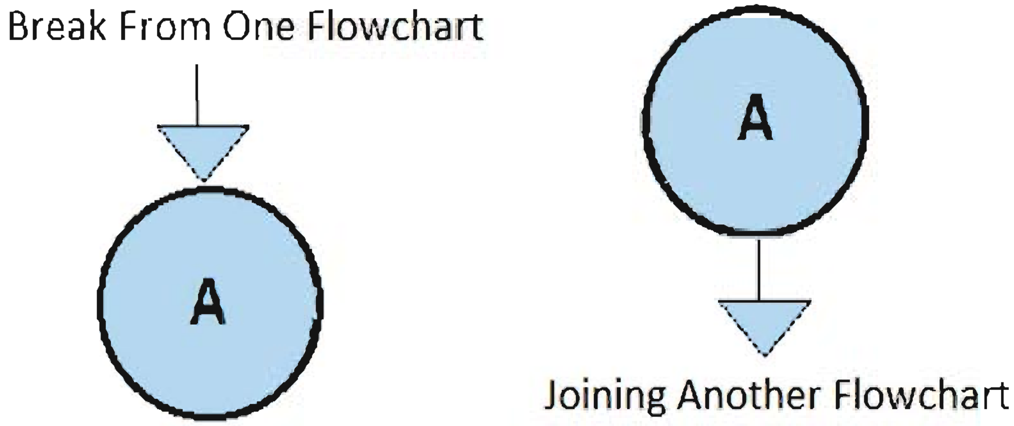
અમુક સમયે નિર્ણય લેવામાં બે કરતાં વધારે વિકલ્પોની આપણને જરૂર પડે છે. આવા સમયે આકૃતિ 9.6માં દર્શાવ્યા પ્રમાણે એક કરતાં વધારે નિર્ણયપેટીઓને જોડીને જરૂરી વિકલ્પ બનાવવમાં આવે છે.



આકૃતિ 9.6 : એક કરતાં વધારે નિર્ણયોનું જોડાણ

આકૃતિ 9.6માં દર્શાવ્યા પ્રમાણે સૌપ્રથમ શરત તપાસવામાં આવે છે અને જો તેનું પરિણામ ખરું આવે તો પહેલાં વિધાનોના ગણનો અમલ થશે. જો પરિણામ ખોટું આવે તો બીજી શરત તપાસવામાં આવશે. આ પ્રક્રિયા આપેલી બધી શરતો માટે ફરીફરીને કરવામાં આવશે. અહીં એ નોંધ કરો કે જ્યારે કોઈ એક શરતનું પરિણામ ખરું આવે તો તે પછીની શરત તપાસવામાં આવતી નથી.

અનુસંધાન ચિહ્ન Connector : અનુસંધાન ચિહ્ન દર્શાવવા માટે વર્તુળનું ચિહ્ન વપરાય છે. અમુક સમયે એ શક્ય છે કે ફ્લોચાર્ટ એટલો મોટો બની જાય કે તે એક પાનામાં સમાવી ન શકાય અથવા બે પ્રક્રિયાને તીર વડે જોડવી શક્ય ન બને. આવી પરિસ્થિતિમાં બે ભાગને જોડવા માટે અનુસંધાન ચિહ્ન (કનેક્ટર) વાપરવામાં આવે છે. જોડાણ માટે ઓછામાં ઓછા બે વર્તુળ જરૂરી છે; આકૃતિ 9.7માં દર્શાવ્યા પ્રમાણે એક તીર વર્તુળની અંદર જતું અને બીજું તીર વર્તુળમાંથી બહાર આવતું. જુદા જુદા અનુસંધાન ચિહ્ન ઓળખવા માટે અલગ અલગ અક્ષરો જોડીમાં વપરાય છે.



આકૃતિ 9.7 : અનુસંધાન ચિહ્ન (કનેક્ટર)

ફ્લોચાર્ટનાં ઉદાહરણો :

અત્યાર સુધીમાં આપણે જાણ્યું કે ફ્લોચાર્ટ શું છે અને તેમાં વપરાતા ચિહ્નો બાબત જાણકારી મેળવી. ચાલો, હવે આપણે ફ્લોચાર્ટ બનાવીને કેટલીક સમસ્યાઓનો ઉકેલ લાવીએ.

ઉદાહરણ 1 : જમીન ઉપર લાદી લગાવવાની કિંમત 10 રૂ. ચો. ફીટ છે. હવે ધારો કે આપણી સમસ્યા લંબચોરસ આકારના નગરના સભાગૃહમાં જમીન ઉપર લાદી લગાવવાની કુલ કિંમત શોધવાની છે.

જવાબ :

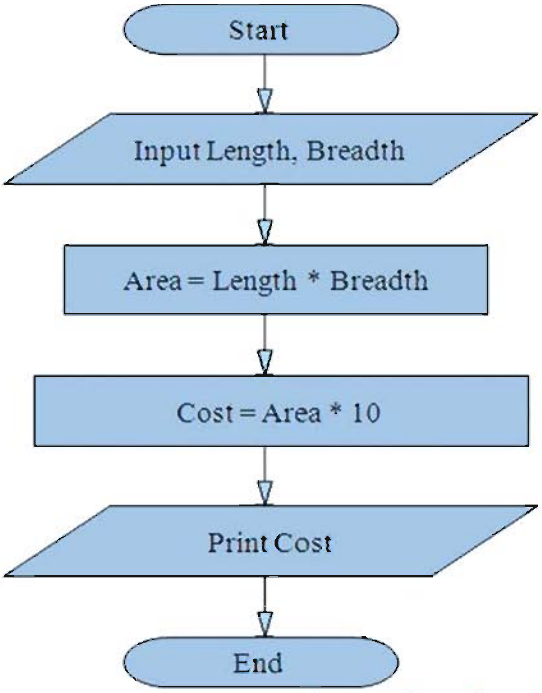
જમીન ઉપર લાદી લગાવવાનો કુલ ખર્ચ શોધવા માટે આપણે પહેલાં જમીનનું ક્ષેત્રફળ શોધવું પડે. આપણે જાણીએ છીએ કે લંબચોરસનું ક્ષેત્રફળ એ તેની લંબાઈ અને પહોળાઈનો ગુણાકાર છે. આથી આપણે જો સભાગૃહની લંબાઈ અને પહોળાઈ જાણતા હોઈએ તો લંબચોરસનું ક્ષેત્રફળ શોધી શકીએ. જમીનનું ક્ષેત્રફળ શોધવા માટે આ મુજબ સૂત્ર આપી શકાય : ક્ષેત્રફળ = લંબાઈ × પહોળાઈ

હવે જમીન ઉપર લાદી લગાવવાની કુલ કિંમત શોધવા માટે આપણે જમીનનું ક્ષેત્રફળ અને એક ચો. ફીટ લગાવવાની કિંમતનો ગુણાકાર કરવો પડે. કુલ કિંમત શોધવાનું સૂત્ર : કુલ કિંમત = ક્ષેત્રફળ × 10.

અહીં સમસ્યાના ઉકેલ માટે આપણે ચાર ચલ ક્ષેત્રફળ, લંબાઈ, પહોળાઈ અને કિંમતની જરૂર પડે છે. હજી આગળ વધતાં પહેલાં આપણે પહેલા શબ્દ ચલની વ્યાખ્યા કરીએ, કારણ કે તે બધા ફ્લોચાર્ટમાં વપરાય છે.

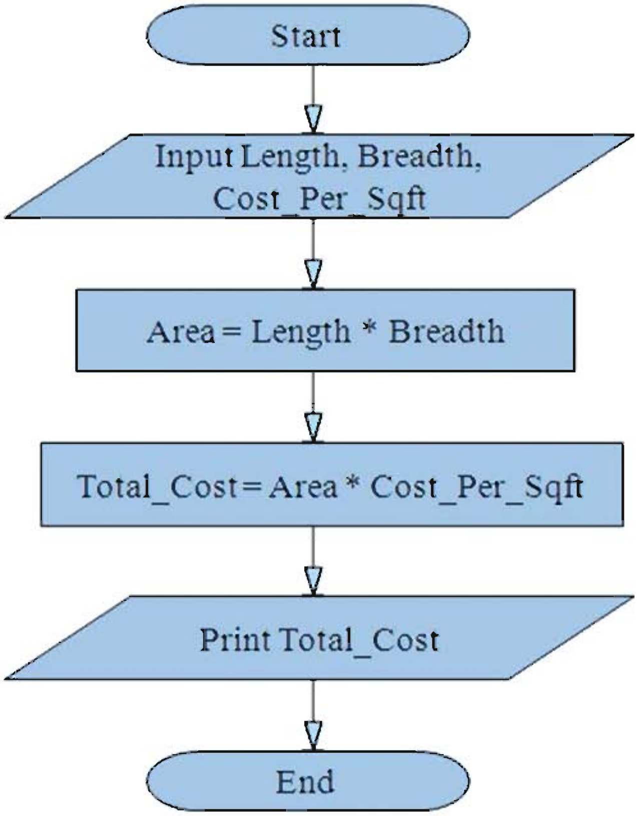
ચલ (Variable) : ચલ એ એક એન્ટિટી (entity) છે જેની કિંમત પ્રક્રિયાના અમલ દરમિયાન બદલી શકાય છે અને તેને કિંમત આપી શકાય છે.

જમીન ઉપર લાદી લગાવવાની કિંમત શોધવાનો ફ્લોચાર્ટ આકૃતિ 9.8માં દર્શાવ્યો છે.



આકૃતિ 9.8 : લાદી લગાવવાની કિંમત શોધવાનો ફ્લોચાર્ટ

પદાવલિ કિંમત = ક્ષેત્રફળ × 10માં કિંમત 10 જુઓ, અહીં 10 એ અચળ (કિંમત) કહેવાય છે. આ એક એવી એન્ટિટી છે, જેની કિંમત એક વાર આપીએ તે પછી આખી પ્રક્રિયા દરમિયાન તેમાં ફેરફાર થતો નથી અને તે સ્થાયી રહે છે. આકૃતિ 9.8માં દર્શાવેલા ફ્લોચાર્ટ દ્વારા આપણે જે ઉકેલ મેળવીએ છીએ તે ફક્ત એ કિસ્સામાં જ વાપરી શકાય જ્યારે એક ચો. ફીટ લાદી લગાવવાની કિંમત રૂ. 10 હોય. જો લાદી લગાવવાની કિંમતમાં ફેરફાર થાય તો ઉપર જણાવેલો ઉકેલ ન ચાલે. ઉપર જણાવેલી સમસ્યાનો સર્વસામાન્ય ઉકેલ આકૃતિ 9.9માં આપેલો છે.



આકૃતિ 9.9 : લાદી લગાવવાનો ખર્ચ શોધવાનો સુધારાયેલ ફ્લોચાર્ટ

આકૃતિ 9.9નું નિરીક્ષણ કરો, અહીં આપણે ચારને બદલે પાંચ ચલનો ઉપયોગ કર્યો છે. અહીં Cost_Per_Sqft ચલનો ઉપયોગ એક ચો. ફીટ લાદી લગાવવાની કિંમતનો સંગ્રહ કરવા માટે કર્યો છે. દરેક સમયે આપણે તેની કિંમતમાં ફેરફાર કરી શકીએ અને આપણે જરૂરી આઉટપુટ મેળવી શકીએ. કોષ્ટક 9.1માં આવાં કેટલાક આઉટપુટ આપેલાં છે.

Length	Breadth	Area=Length*Breadth	Cost_Per_Sqft	Total_Cost=Area*Cost_Per_Sqft
50	50	2500	10	25000
25	75	1875	10	18750
45	35	1575	20	31500

કોષ્ટક 9.1 : આકૃતિ 9.9માં દર્શાવેલા ફ્લોચાર્ટનું આઉટપુટ

ઉદાહરણ 2 : ધારો કે તમારા શહેરનાં રમતગમત સંકુલમાં એક સુંદર અને વર્તુળ ક્રિકેટનું મેદાન છે. તેના સત્તાધીશો આ મેદાનની ફરતે વાડ બનાવવા ઇચ્છે છે. આ ઉપરાંત તેઓ સંપૂર્ણ મેદાનને લૉન (વ્યવસ્થિત રીતે કાપેલા અને સુંવાળા ઘાસવાળી હરિયાળી જમીન)થી ઢાંકવા ઇચ્છે છે. સત્તાધીશો જાણવા માગે છે કે કેટલા મીટરની વાડ જોઈશે ? તેઓ એ પણ જાણવા ઇચ્છે છે કે મેદાનનું કુલ ક્ષેત્રફળ કેટલું છે કે જેને લૉન વડે ઢાંકવાની જરૂર છે.

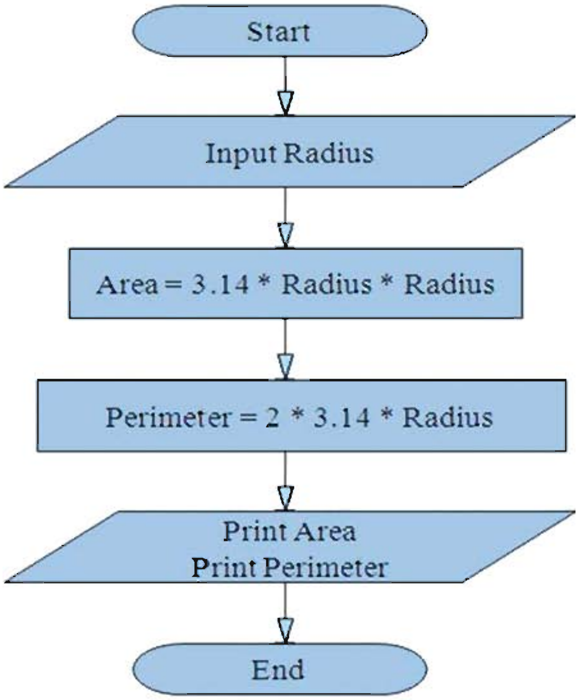
જવાબ :

આ પ્રશ્નના ઉકેલ માટે આપણે મેદાનનું ક્ષેત્રફળ અને તેની પરિમિતિ શોધવી પડશે. ગોળ મેદાનનું ક્ષેત્રફળ શોધવા માટેનું સૂત્ર ક્ષેત્રફળ $= \pi * R^2$ અને મેદાનની પરિમિતિ $= 2 * \pi * R$ (અહીં πની કિંમત 3.14 અને R એ મેદાનની ત્રિજ્યા છે)

આથી, આ પ્રશ્નના ઉકેલ માટે ત્રણ ચલ Radius (ત્રિજ્યા), Area (ક્ષેત્રફળ) અને Perimeter (પરિમિતિ) આપણને જરૂર પડશે. ફ્લોચાર્ટમાં આપણે ક્ષેત્રફળની ગણતરી અને મેદાનની પરિમિતિ શોધવા માટે જે સૂત્ર વાપરીશું તે નીચે આપેલાં છે.

Area = 3.14 * Radius * Radius (ક્ષેત્રફળ = 3.14 * ત્રિજ્યા * ત્રિજ્યા)

અને Perimeter = 2 * 3.14 * Radius (પરિમિતિ = 2 * 3.14 * ત્રિજ્યા)

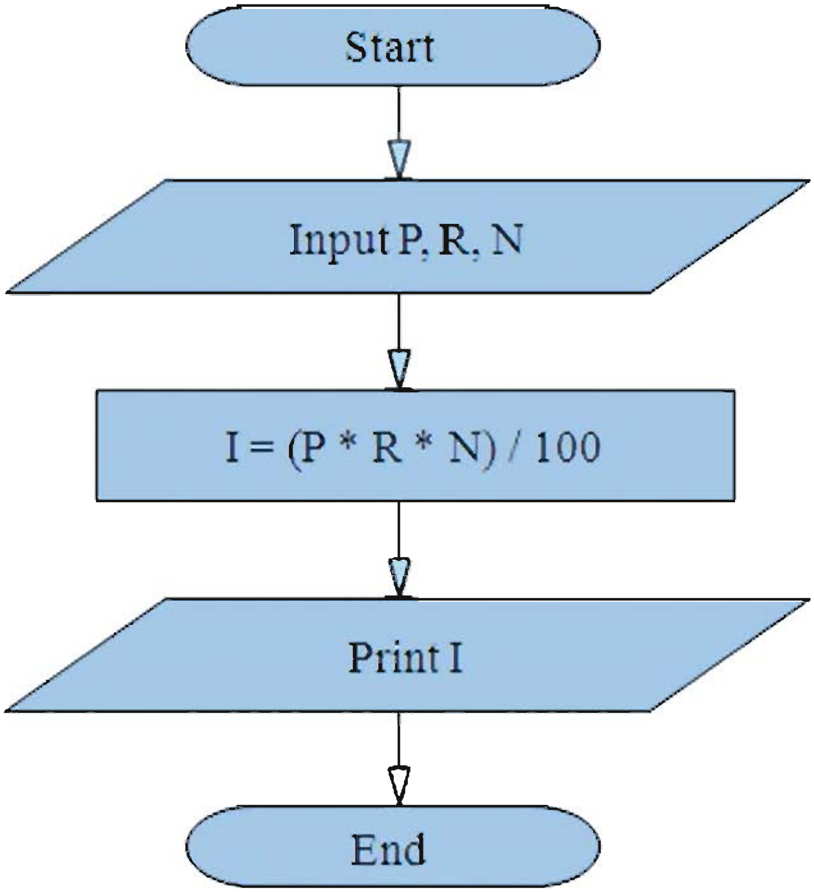


આકૃતિ 9.10 : વર્તુળનું ક્ષેત્રફળ અને પરિમિતિ શોધવાનો ફ્લોચાર્ટ

ઉદાહરણ 3 : જહાનવીએ બૅન્કમાંથી 6 વર્ષની 11.5 ટકાના દરથી 35,000 રૂ.ની લોન લીધી. તેણીએ બૅન્કને કેટલું સાદું વ્યાજ ચૂકવવું પડશે તેની ગણતરી કરો.

જવાબ

કોઈ લોનની મુદલ રકમ 'P', વ્યાજનો દર 'R' અને 'N' વર્ષ માટેના સાદા વ્યાજ 'I' ની ગણતરી માટેનું સૂત્ર **I = (P*R*N)/100** છે. આ પ્રશ્નના ઉકેલ માટે આપણે ચાર ચલ I, P, R, N અને અચળ કિંમત 100ની જરૂર પડશે. આ પ્રશ્નનો ઉકેલ આકૃતિ 9.11માં ફ્લોચાર્ટના રૂપમાં દર્શાવ્યો છે.



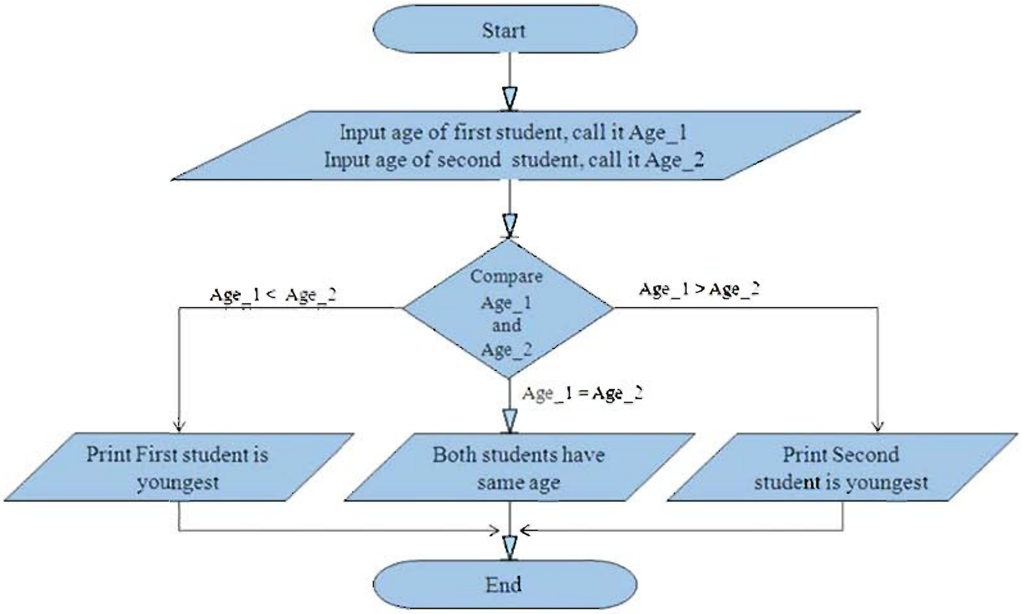
આકૃતિ 9.11 : સાદા વ્યાજની ગણતરી માટેનો ફ્લોચાર્ટ

ઉદાહરણ 4 : ધારો કે બે વિદ્યાર્થીઓમાંથી તમારે નાની ઉંમરનો વિદ્યાર્થી શોધવો છે, આ કિસ્સામાં વિદ્યાર્થીની ઉંમર એ નિવેશ (ઇનપુટ) હશે.

જવાબ :

ઉપરના પ્રશ્નનો ઉકેલ નીચે જણાવેલી ક્રિયાઓ કરવાથી મળશે :

સૌપ્રથમ બંને વિદ્યાર્થીઓની ઉંમરનું ઇનપુટ (નિવેશ) લો. તે ઉંમર Age_1 અને Age_2 ચલમાં રાખો. હવે Age_1 અને Age_2ની કિંમતની સરખામણી કરો. જો બંને કિંમત સરખી હોય તો આપણી પાસે સરખી ઉંમરના બંને વિદ્યાર્થી છે અને આથી બંને નાનામાં નાના છે. જો Age_1ની કિંમત Age_2 કરતાં ઓછી હોય તો પહેલો વિદ્યાર્થી નાનો છે, નહીંતર બીજો વિદ્યાર્થી. આ પ્રશ્નના ઉકેલનો ફ્લોચાર્ટ આકૃતિ 9.12માં આપેલો છે.



આકૃતિ 9.12 : બે વિદ્યાર્થીઓમાંથી નાનો વિદ્યાર્થી શોધવાનો ફ્લોચાર્ટ

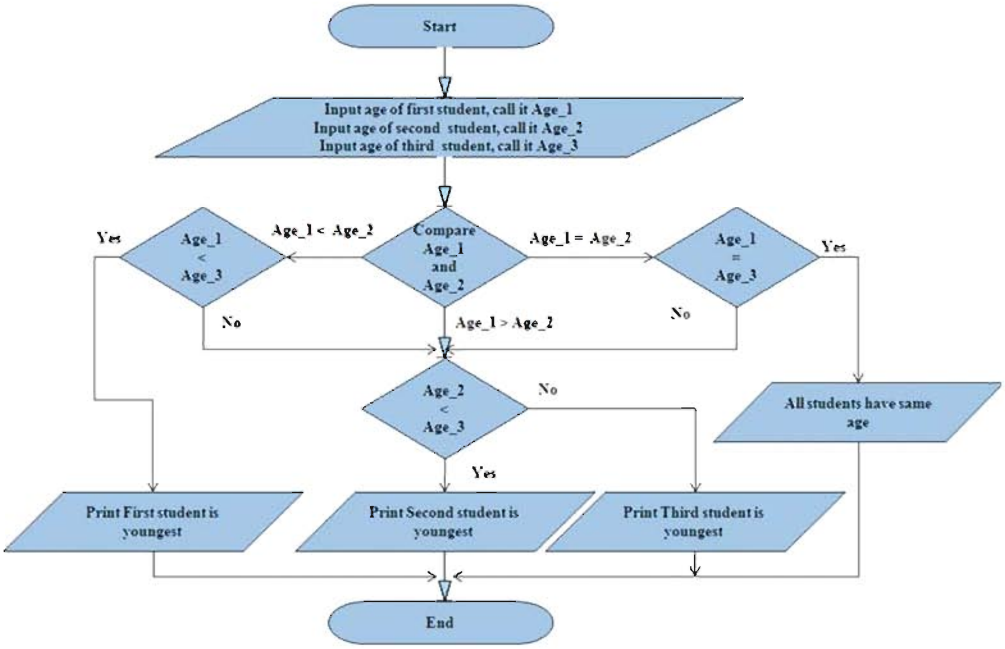
ઉદાહરણ 5 : હવે આપણે ત્રણ વિદ્યાર્થીઓમાંથી સૌથી નાનો વિદ્યાર્થી શોધવાનો પ્રયત્ન કરીએ.

જવાબ :

સૌપ્રથમ ત્રણ વિદ્યાર્થીઓની ઉંમરનું ઇનપુટ લો અને તે ચલ Age_1, Age_2 અને Age_3માં રાખો. જો ત્રણે કિંમત એકસરખી હોય તો આપણી પાસે ત્રણે સરખી ઉંમરના વિદ્યાર્થી છે. આથી ત્રણે વિદ્યાર્થીઓને નાનામાં નાના ગણવામાં આવે.

જો એમ ન હોય તો Age_1 અને Age_2ની સરખામણી કરો. જો Age_1ની કિંમત Age_2 કરતાં ઓછી હોય તો Age_1 ને Age_3 સાથે સરખાવો અને હજી Age_1 એ Age_3 કરતાં ઓછી હોય તો પહેલો વિદ્યાર્થી નાનામાં નાનો છે.

ઉપર જે સરખામણી કરી તેમાં જો Age_2 એ Age_1 કરતાં ઓછી હોય તો Age_2ને Age_3 સાથે સરખાવો અને જો Age_2 એ Age_3 કરતાં ઓછી હોય તો બીજો વિદ્યાર્થી નાનામાં નાનો છે. આ સિવાય ત્રીજો વિદ્યાર્થી નાનામાં નાનો છે. આ પ્રશ્નના ઉકેલનો ફ્લોચાર્ટ આકૃતિ 9.13માં આપેલો છે.



આકૃતિ 9.13 : ત્રણ વિદ્યાર્થીઓમાંથી નાનામાં નાનો વિદ્યાર્થી શોધવાનો ફ્લોચાર્ટ

ઉદાહરણ 6 : ચાલો, ઉપરના ઉદાહરણમાં હજી પણ સુધારો કરીએ અને વર્ગના બધા વિદ્યાર્થીઓમાંથી નાનામાં નાનો વિદ્યાર્થી શોધીએ. આ પ્રશ્નના ઉકેલ માટે નીચે જણાવેલાં પગલાંનો ઉપયોગ કરી શકાય :

પગલું 1 : પહેલો વિદ્યાર્થી લો અને તેની કોઈ પણ ઉંમર હોય, ધારો કે તે MINIMUM છે.

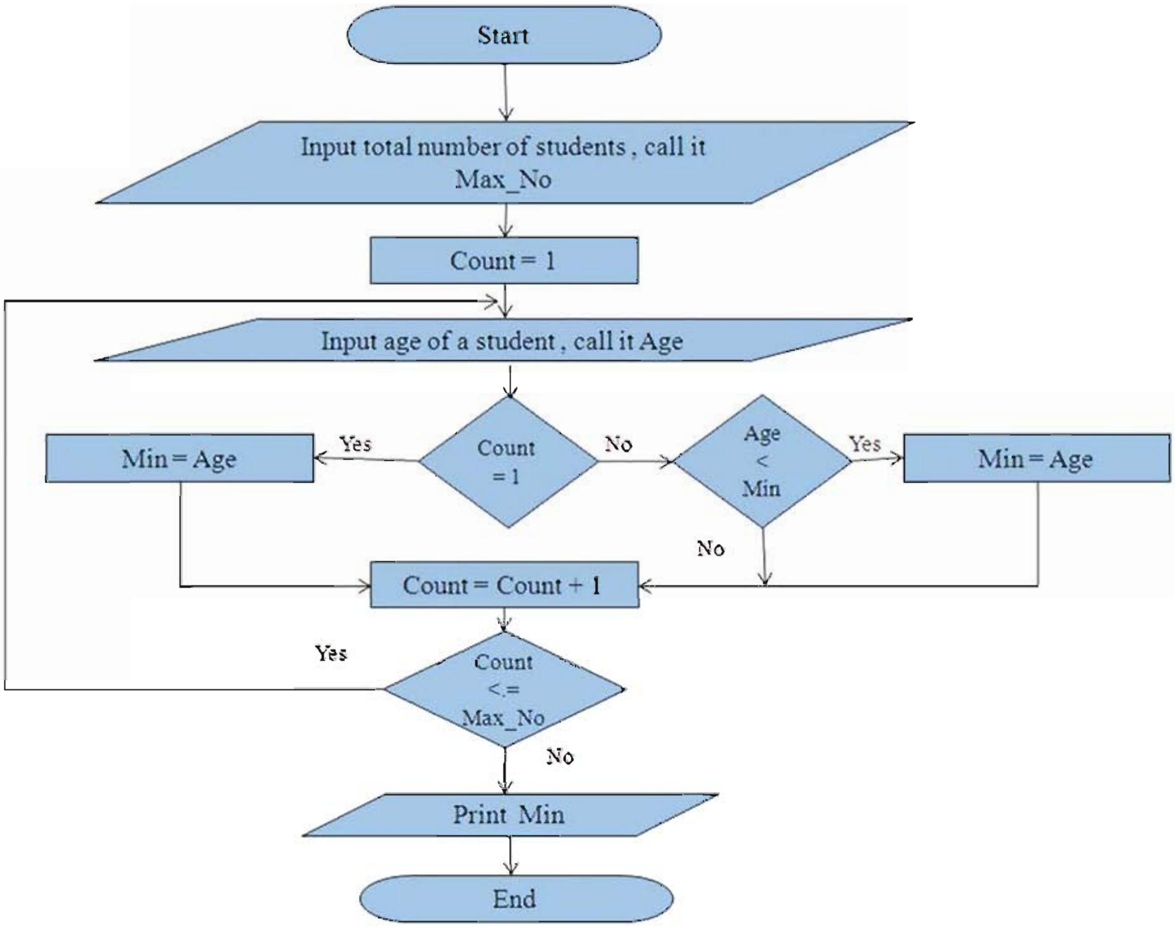
પગલું 2 : હવે બીજો વિદ્યાર્થી પસંદ કરો અને તેની ઉંમર MINIMUM જણાવેલ તેની સાથે સરખાવો.

પગલું 3 : જો નવા વિદ્યાર્થીની ઉંમર ઓછી હોય તો તેની ઉંમર MINIMUM બનાવીશું અને અગાઉના વિદ્યાર્થીની ઉંમર રદ કરીશું.

પગલું 4 : જો બંને વિદ્યાર્થીઓની ઉંમર એકસરખી હોય તો નવા વિદ્યાર્થીની ઉંમરની અવગણના કરવામાં આવશે અને અગાઉના વિદ્યાર્થીની ઉંમર હજી પણ MINIMUM છે.

પગલું 5 : આપણે ઉપર જણાવેલાં પગલાં 3 અને 4નું પુનરાવર્તન ચાલુ રાખીશું કે જ્યાં સુધી બધા વિદ્યાર્થીઓમાંથી નાનામાં નાનો વિદ્યાર્થી આપણને મળી જાય. જ્યારે અમુક વિધાનોને એક કરતાં વધારે વખત પુનરાવર્તિત કરવામાં આવે તેને લૂપ (Loop) કહેવામાં આવે છે.

આ પ્રશ્નના ઉકેલનો ફ્લોચાર્ટ આકૃતિ 9.14માં આપેલો છે.



આકૃતિ 9.14 : વર્ગના બધા વિદ્યાર્થીઓમાંથી સૌથી યુવાન શોધવાનો ફ્લોચાર્ટ

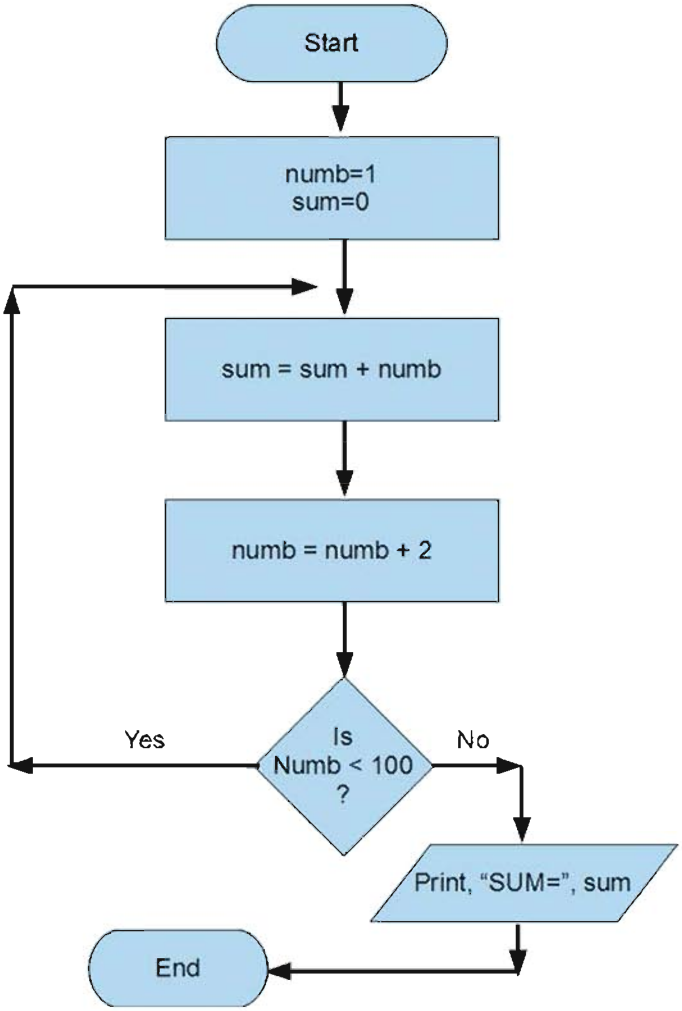
ઉદાહરણ 7 : ચાલો, હવે આપણે એક ગણિતનો પ્રશ્ન ઉકેલીએ. ધારો કે આપણે પહેલી 50 એકી સંખ્યા (odd numbers) ઓનો સરવાળો કરવા ઇચ્છીએ છીએ. એટલે કે આપણે 1 + 3 + 5 + 7 + ... + 99નો સરવાળો કરવા ઇચ્છીએ છીએ.

જવાબ :

ધારો કે તમારી પાસે બે બૉક્સ (પેટીઓ) છે, એક ડાબા હાથમાં અને બીજી જમણા હાથમાં. હવે ડાબી બાજુના બૉક્સમાં એક લખોટી મૂકો અને હવે તે બૉક્સને જમણા હાથમાં રહેલા બૉક્સમાં ખાલી કરો. હવે ડાબા હાથમાં રહેલા બૉક્સમાં ત્રણ લખોટી મૂકો અને તેને જમણા હાથમાં રહેલા બૉક્સમાં ખાલી કરો (બદલી કરો). આથી જમણા હાથમાં રહેલા બૉક્સમાં

કેટલી થશે ? જમણા હાથમાં રહેલા બૉક્સમાં હવે કુલ 1 + 3 લખોટીઓ હશે. જો આપણે આ પ્રક્રિયા 5, 7, ... 99 લખોટીઓ સાથે ચાલુ રાખીશું તો જમણા હાથમાં રહેલા બૉક્સમાં 1 + 3 + 5 + 7 + ... + 99 લખોટીઓ હશે. અહીં એ નોંધ કરો કે આપણે ધારેલું કે જમણા હાથનું બૉક્સ શરૂઆતમાં ખાલી છે.

આ ક્રિયાને અનુરૂપ, આપણે બે ચલ numb અને sum લઈએ. numb ચલની શરૂઆતની કિંમત 1 અને sumની કિંમત 0 (ખાલી બૉક્સની જેમ) રાખીએ. હવે લૂપના દરેક સમયે numb ચલની કિંમત 2થી વધારવામાં આવે છે અને તેની કિંમત sumમાં ઉમેરવામાં આવે છે. આથી numb ચલની કિંમત બદલાતી રહેશે અને તે 1, 3, 5, ... અને છેલ્લે 99 બનશે. આથી લૂપનો અંત આવે ત્યાં સુધી વિધાન sum = sum + numbથી numbની કિંમત sumમાં ઉમેરાતી રહેશે. આ પ્રશ્નનો ફ્લોચાર્ટ આકૃતિ 9.15માં આપેલો છે.



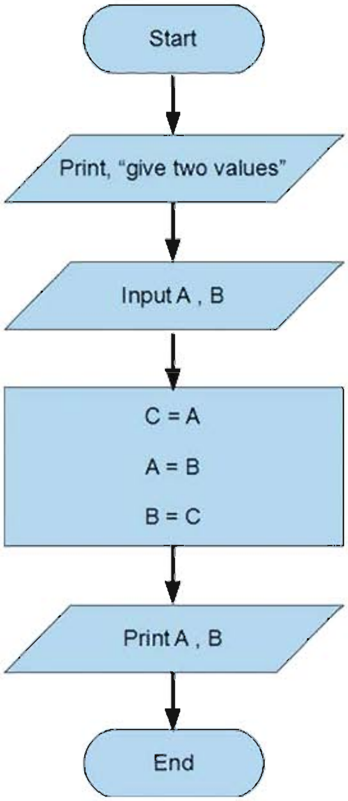
આકૃતિ 9.15 : 1થી 99 સુધીની એકી સંખ્યાઓના સરવાળાનો ફ્લોચાર્ટ

ઉદાહરણ 8 : હવે આપણે બે ચલની કિંમતની અદલાબદલી કરવાની બે જુદી જુદી રીતો જોઈએ.

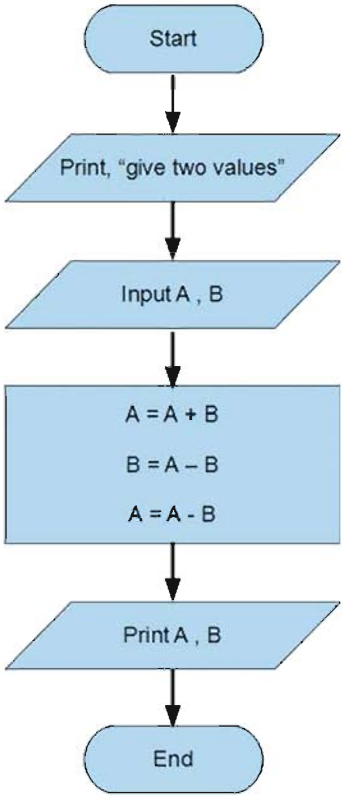
જવાબ :

બે ચલ A અને B લો. જો A અથવા Bને કોઈ અન્ય કિંમત વડે બદલવામાં આવશે તો તેમાં શરૂઆતમાં રહેલી કિંમત આપણે ગુમાવીશું. આથી આપણે એક વધારાનો ચલ Cનો ઉપયોગ કરીશું. Aની કિંમત જળવાઈ રહે તે માટે તેની કિંમત પહેલાં આપણે Cમાં રાખીએ. તે પછી Bની કિંમત આપણે Aમાં લઈ જઈએ અને ત્યાર બાદ Aની સાચવેલી કિંમત (જે Cમાં રાખેલી છે) Bમાં લઈ જઈએ. આકૃતિ 9.16માં આ ઉકેલનો ફ્લોચાર્ટ આપેલો છે.

આકૃતિ 9.17માં આ જ પ્રશ્નના ઉકેલનો ફ્લોચાર્ટ આપેલો છે, પણ તેમાં વધારાના ચલનો ઉપયોગ થતો નથી. ખરેખર કિંમત બદલવા માટે સરવાળા અને બાદબાકીનો ઉપયોગ કરેલો છે.



આકૃતિ 9.16 : વધારાના ચલનો ઉપયોગ કરીને કિંમતની અદલાબદલી



આકૃતિ 9.17 : વધારાના ચલ વિના કિંમતની અદલાબદલી

ફ્લોચાર્ટ(Flowchart)ના ફાયદાઓ અને ગેરફાયદાઓ (Advantages and Disadvantages of Flowchart)

જ્યારે આપણે કોઈ પણ પ્રકારનાં ટૂલ્સ (tools) વાપરીએ છીએ, ત્યારે તેના કેટલાક ફાયદાઓ અને ગેરફાયદાઓ હોય છે. ફ્લોચાર્ટ તેમાં અપવાદ નથી. તેના કેટલાક ફાયદાઓ અને મર્યાદાઓ છે.

ફ્લોચાર્ટના ફાયદા :

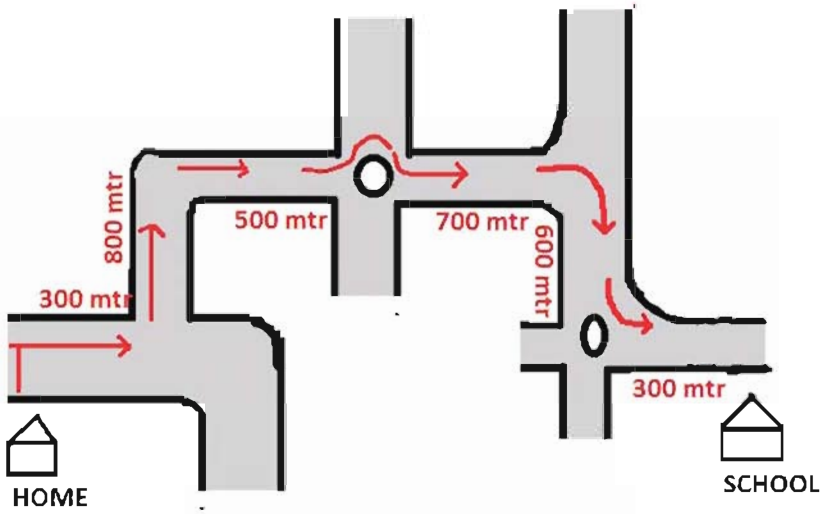
1. તે સમસ્યાનો ઉકેલ ચિત્રાત્મક સ્વરૂપે રજૂ કરતો હોવાથી સમજવો ખૂબ સરળ છે.
2. તે કમ્પ્યૂટરની સૂચનાઓ લખવામાં ખૂબ જ અનુકૂળ મદદ પૂરી પાડે છે.
3. સમસ્યાના ઉકેલના પગલાંની સમીક્ષા કરવામાં અને સુધારવામાં પ્રોગ્રામરને સહાયભૂત બને છે.
4. સમસ્યાના ઉકેલની જુદી જુદી રીતો વિશે ચર્ચા કરવામાં તે મદદરૂપ બને છે, જેને કારણે અંતે તેની સરખામણી ભૂલરહિત કરવાની સગવડ તે પૂરી પાડે છે.

ફ્લોચાર્ટના ગેરફાયદા :

1. મોટી અને ગૂંચવાડાભારી સમસ્યાનો ફ્લોચાર્ટ દોરવાનું કામ ઘણો સમય માગી લે છે અને તે થકાવટવાળું કામ છે.
2. ફ્લોચાર્ટમાં અનેક ચિહ્નો હોય છે અને આથી પ્રોગ્રામના તર્કમાં કોઈ પણ ફેરફાર કરવાથી અનેક સમયે નવો ફ્લોચાર્ટ દોરવાની જરૂર પડે છે.
3. ફ્લોચાર્ટમાં ડેટા કેટલા વિગતવાર જણાવવા તેના કોઈ ધોરણ નથી. આથી એકસરખા તર્કવાળી સમસ્યાના ફ્લોચાર્ટ તેની અંદરની માહિતીથી અલગ પડે છે.

અલગોરિધમ (Algorithm) (કલનવિધિ)

તમારો મિત્ર તમારા ઘેર આવી ગયો છે અને હવે તમારી નિશાળમાં આવવાની ઇચ્છા રાખે છે. તે તમને ફોન કરે છે અને તમારી નિશાળ સુધી પહોંચવાની દિશા પૂછે છે. સ્કૂલ કઈ રીતે પહોંચી શકાય તેની સૂચના આપવા માટે તમારા મગજમાં પહેલાં કોઈ નિયત માર્ગ હોય એ જરૂરી છે. ધારો કે, આકૃતિ 9.18માં દર્શાવ્યા પ્રમાણેનો નિયત માર્ગનો નકશો તેને મદદ કરવા માટે છે.



આકૃતિ 9.18 : નિયત માર્ગનો નકશો

જવાબ :

આકૃતિ 9.18માં આપેલાં નિયત માર્ગના નકશાનો ઉપયોગ કરીને તમે તમારા મિત્રને નીચે જણાવ્યા પ્રમાણે એક પછી એક પગલાંનું માર્ગદર્શન આપી શકશો :

પગલું 1 : મારા ઘરથી શરૂ કરી પહેલાં જમણી બાજુ ફરો અને 300 મીટર સીધા ચાલો.

પગલું 2 : હવે ડાબી બાજુ ફરો અને 800 મીટર સીધા જાઓ.

પગલું 3 : હવે જમણી બાજુ ફરો અને 500 મીટર ચાલો આથી ચાર રસ્તા પર એક સર્કલ આવશે.

પગલું 4 : સર્કલ પર વળાંક ન લો અને સીધા એ જ રસ્તા પર બીજા 700 મીટર ચાલો જેના પર તમે ચાલતા આવ્યા હતા.

પગલું 5 : હવે જમણી બાજુ ફરો અને 600 મીટર ચાલ્યા બાદ ફરી ચાર રસ્તા વચ્ચે અંડાકાર આકારના ફુવારા પર તમે પહોંચશો.

પગલું 6 : આ જંકશનથી ડાબી બાજુ ફરો અને 300 મીટર ચાલો.

પગલું 7 : હવે તમે શેરીની જમણી બાજુએ મારી નિશાળ જોઈ શકશો.

ઉપરનાં ઉદાહરણ પરથી આપણે કહી શકીએ કે પગલાંઓની શ્રેણીને અનુસરવાથી તમારી નિશાળે પહોંચવાનો પ્રશ્ન ઉકેલી શકીએ છીએ. કમ્પ્યૂટર સાયન્સમાં અલગોરિધમ કોઈ ચોક્કસ સમસ્યાના ઉકેલ માટેની ક્રમશઃ કાર્યપ્રણાલીનો નિર્દેશ કરે છે. અલગોરિધમ અંગ્રેજી જેવી પ્રાકૃતિક ભાષામાં લખવામાં આવે છે. એક વાર અલગોરિધમ લખવામાં આવે પછી તેના ઉપરથી કમ્પ્યૂટર પ્રોગ્રામ લખવાનું અતિ સરળ બની જાય છે. ચાલો, હવે થોડાક અલગોરિધમ લખવાનું શીખીએ.

ઉદાહરણ 1 :

1થી 100ની અંદર 11 વડે વિભાજિત સંખ્યાઓનો સરવાળો શોધવાનો અલગોરિધમ લખો.

અલગોરિધમ :

પગલું 1 : આરંભ

પગલું 2 : Num અને Sum બે ચલ લો.

પગલું 3 : શરૂઆતમાં કિંમત આપો. Num = 1 અને Sum = 0.

પગલું 4 : Num ચલ 11 વડે ભાગી શકાય તેમ છે કે કેમ તે તપાસો. આ માટે Numને 11 વડે ભાગીને શેષ શોધવી પડશે. આ શેષને અન્ય ચલ (ધારો કે) Rમાં સંગ્રહ કરો.

R = Num – int(Num/11) * 11

પગલું 5 : જો Rની કિંમત શૂન્ય ન હોય તો પગલું 7 પર જાઓ.

પગલું 6 : Sum = Sum + Num

પગલું 7 : Num = Num + 1

પગલું 8 : જો Num ની કિંમત 100 થી ઓછી હોય તો પગલું-4 પર જાઓ.

પગલું 9 : Sum પ્રિન્ટ કરો.

પગલું 10 : અંત

સમજૂતી :

શેષ શોધવા માટે આપણે Numમાં સંગ્રહ કરેલી સંખ્યાને 11 વડે ભાગીએ છીએ અને પછી તેનો પૂર્ણાંક ભાગ લઈએ છીએ. દા.ત. જો Numની કિંમત 39 હોય તો તેને 11 વડે ભાગવાથી આપણે 3.54545455 મેળવીએ. જો જવાબનો પૂર્ણાંક ભાગ લઈએ તો તે ફક્ત 3 છે. દશાંશ ચિહ્ન પછીનો ભાગ કાઢી નાખવામાં આવે છે. જો પૂર્ણાંક ભાગને 11 સાથે ગુણાકાર કરવામાં આવે તો આપણે 33 મેળવીશું. જો આ 33ને મૂળ સંખ્યા 39માંથી બાદ કરવામાં આવે તો આપણે કિંમત 6 મેળવીશું. આ કિંમત શેષ છે. અહીં એ નોંધ કરશો કે int નામની પ્રક્રિયા કે જે સંખ્યાનો પૂર્ણાંક ભાગ આપે છે, તેની આપણે ધારણા કરેલી છે.

પગલાં 1થી 4 જાતે સમજૂતી આપે તેવાં છે. પગલું 5 ઉપર જણાવેલી તક્નિક વાપરીને શેષની કિંમત તપાસે છે. જો શેષ શૂન્ય ન હોય તો Numની કિંમત 11 વડે વિભાજિત નથી અને આપણે તે પછીની સંખ્યા લઈશું. જો Numની કિંમત 11 વડે વિભાજિત હોય તો આપણે પગલાં 6 ઉપર જઈશું. આપણે Numની કિંમત Sumમાં ઉમેરીશું અને પછી પગલાં 7નો અમલ કરીશું. પગલાં 8માં આપણે Num કિંમત 100 છે કે ઓછી તેની તપાસ કરીશું. આ તપાસ લૂપની બહાર નીકળવા માટે વપરાય છે. આપણે આ રીતે જ્યાં સુધી Num ની કિંમત 100 કરતાં ઓછી હશે ત્યાં સુધી ઉપરની ક્રિયા ચાલુ રાખીશું. અંતમાં આપણે Sum ની કિંમત છાપીશું.

ઉદાહરણ 2 : આપેલી મુદ્દલ રકમ, સમયગાળો અને વ્યાજ દર પરથી વાર્ષિક વ્યાજ ગણવામાં આવે ત્યારે ચક્રવૃદ્ધિ વ્યાજની ગણતરી કરવા માટેનો અલગોરિધમ લખો.

અલગોરિધમ :

આપણે ચક્રવૃદ્ધિ વ્યાજની ગણતરી માટે નીચેનું સૂત્ર વાપરીએ છીએ :

$$CI = P * (1 + R/100)^N - P$$

અહીં P-મુદ્દલ રકમ, R-વ્યાજનો દર અને N-વર્ષમાં સમયગાળો

પગલું 1 : આરંભ

પગલું 2 : ચાર ચલ P, N, R અને CI લો.

પગલું 3 : P, N અને Rની કિંમત આપો.

પગલું 4 : સૂત્ર $CI = P * (1 + R/100)^N - P$ વાપરી ચક્રવૃદ્ધિ વ્યાજ ગણો.

પગલું 5 : CIમાં સંગ્રહ કરેલી કિંમત છાપો.

પગલું 6 : અંત

ઉદાહરણ 3 :

કોઈ કર્મચારીના કામના કલાકોના આધારે અઠવાડિક પગાર શોધો. આ પ્રશ્નના જવાબ માટે આપણે કર્મચારીને એક કલાકનો કેટલો પગાર ચૂકવાય છે તે જાણવું પડે.

અલગોરિધમ :

અઠવાડિક પગાર ચૂકવવા માટે આપણે બે કિંમતના ઇનપુટ (નિવેશ) લેવા પડે : કર્મચારીએ કેટલાં કલાક કામ કર્યું તે અને દરેક કલાકનો પગાર.

પગલું-1 : આરંભ

પગલું-2 : ત્રણ ચલ No_Of_Hrs_Worked, Pay_Per_Hour અને Weekly_Pay લો.

પગલું-3 : No_Of_Hrs_Worked અને Pay_Per_Hourની કિંમત આપો.

પગલું-4 : નીચેનું સૂત્ર વાપરી અઠવાડિક પગારની ગણતરી કરો :

Weekly_Pay = No_Of_Hrs_Worked * Pay_Per_Hour

પગલું-5 : ચલ Weekly_Payમાં સંગ્રહ કરેલી કિંમત છાપો.

પગલું-6 : અંત

સારાંશ

આ પ્રકરણમાં આપણે સમસ્યાના ઉકેલ માટે બે વધારે વપરાતી તકનિક ફ્લોચાર્ટ અને અલગોરિધમ વિશે શીખ્યા. આપણે ફ્લોચાર્ટમાં વપરાતી વિવિધ સંજ્ઞાઓ વિશે પણ જાણ્યું. સમસ્યાના ઉકેલ માટે ફ્લોચાર્ટ વાપરવો કે અલગોરિધમ તેના નિર્ણય માટે વપરાશકર્તાની જરૂરિયાતો બાબત જાણકારી મેળવી.

સ્વાધ્યાય

1. ફ્લોચાર્ટના મૂળભૂત અંગો ક્યા ક્યા છે ?
2. ક્યા પ્રકારની સમસ્યાનો ઉકેલ કમ્પ્યૂટર વડે લાવી શકાય ?
3. ક્યા પ્રકારનાં કાર્યો એ કમ્પ્યૂટરની પ્રક્રિયાના ભાગ છે ?
4. અંકગણિતનાં કાર્યોમાં ક્યા ક્યા કાર્યોનો સમાવેશ થાય છે ?
5. નિર્ણય પેટી(decision box)નો ઉપયોગ ક્યા પ્રકારનાં કાર્ય દર્શાવવા માટે થાય છે ?
6. ચલને વ્યાખ્યાયિત કરો.
7. અલગોરિધમ એટલે શું ? તે ફ્લોચાર્ટથી કઈ રીતે જુદો પડે છે ?
8. આપેલા વિકલ્પોમાંથી ખરો વિકલ્પ પસંદ કરો :

(1) ફ્લોચાર્ટની શરૂઆત દર્શાવવા માટે કઈ સંજ્ઞાનો ઉપયોગ થાય છે ?

(a)

(b)

(c)

(d)

(2) નીચેનામાંથી શું સમસ્યાના ઉકેલ માટે યોગ્ય ક્રમમાં સૂચનાઓનો નિર્દેશ કરે છે ?

(a) અલગોરિધમ (b) ફ્લોચાર્ટ (c) શ્રેણી (d) રોડમેપ (નકશો)

(3) નીચેનામાંથી કયું ચિહ્ન ફ્લોચાર્ટમાં શરત તપાસવા માટે વપરાય છે ?

(a) હીરો (ડાયમંડ) (b) વર્તુળ (સર્કલ) (c) તીર (એરો) (d) ચોરસ (સ્ક્વેર)

(4) નીચેનામાંથી કયું ચિહ્ન ફ્લોચાર્ટમાં આઉટપુટ દર્શાવવા માટે વપરાય છે ?

(a) ચોરસ (સ્ક્વેર) (b) વર્તુળ (સર્કલ)

(c) સમાંતર ચતુર્ભુજ (પૅરેલેલોગ્રામ) (d) ત્રિકોણ (ટ્રાયઍંગલ)

(5) નીચેનામાંથી કયું ચિહ્ન ફ્લોચાર્ટમાં અંત દર્શાવવા માટે વપરાય છે ?

(a) વર્તુળ (b) હીરો

(c) ગોળ ખૂણાવાળો લંબચોરસ (રાઉન્ડેડ રેક્ટેન્ગલ)(d) ચોરસ

(6) નીચેનામાંથી કયું વિધાન અલગોરિધમ અને ફ્લોચાર્ટનો હેતુ જણાવે છે ?

(a) સ્મૃતિ(મેમરી)ની ક્ષમતા જાણવા

(b) અંક પદ્ધતિનો આધાર (base) ઓળખવા

(c) આઉટપુટ(નિર્ગમ)ને પ્રિન્ટર ઉપર મોકલવું

(d) સમસ્યાને સંપૂર્ણપણે અને સ્પષ્ટપણે જણાવવી

(7) નીચેનામાંથી કઈ સમસ્યાના ઉકેલની તકનિક નથી ?

(a) સુડો કોડ (b) ફ્લોચાર્ટ (c) અલગોરિધમ (d) શ્રેણી

(8) નીચેનામાંથી સમસ્યાના ઉકેલની ચિત્રાત્મક રજૂઆત કઈ છે ?

(a) સુડો કોડ (b) ફ્લોચાર્ટ

(c) અલગોરિધમ (d) કમ્પ્યૂટર પ્રોગ્રામ

(9) ફ્લોચાર્ટમાં તીરનું ચિહ્ન શું બતાવવા માટે વપરાય છે ?

(a) કાર્યના પ્રવાહની દિશા (b) કાર્યનો ક્રમ

(c) કાર્યની શરૂઆત (d) કાર્યની પૂર્ણાહુતિ

(10) નીચેનામાંથી શું કોઈ પણ કાર્યનો મુખ્ય ભાગ (core part) છે ?

(a) ઇનપુટ (નિવેશ) (b) આઉટપુટ (નિર્ગમ) (c) પ્રક્રિયા (d) અલગોરિધમ

(11) નીચેનામાંથી કયું ચિહ્ન પ્રક્રિયા દર્શાવે છે ?

(a) લંબચોરસ (b) ચોરસ (c) વર્તુળ (d) હીરો

(12) નીચેનામાંથી ફ્લોચાર્ટમાં જુદા જુદા અનુસંધાન ચિહ્નની જોડી માટે શું વપરાય છે ?

(a) તીર (b) મૂળાક્ષરો કે બીજા અક્ષર

(c) વર્તુળ (d) હીરાનું ચિહ્ન

પ્રાયોગિક સ્વાધ્યાય

નીચેનાં કાર્યો માટે ફ્લોચાર્ટ દોરો અને અલગોરિધમ લખો :

1. આપેલા મીટરને સેન્ટિમીટરમાં ફેરવો.
2. આપેલા સેન્ટિગ્રેડને ફેરનહીટમાં ફેરવો.
3. આપેલી બે સંખ્યામાંથી નાની સંખ્યા શોધો.
4. આપેલી બે સંખ્યામાંથી નાની સંખ્યા શોધો. એ એકી સંખ્યા છે કે બેકી સંખ્યા તે તપાસો.
5. આપેલી ત્રણ સંખ્યામાંથી લઘુતમ સંખ્યા શોધો.
6. આપેલી કોઈ સંખ્યા અન્ય કોઈ આપેલી સંખ્યાથી વિભાજિત છે કે કેમ તે તપાસો.
7. ત્રણ સંખ્યાની સરેરાશ શોધો.
8. કોઈ વિદ્યાર્થીના કુલ ગુણને આધારે શોધો કે તે વિદ્યાર્થી પાસ છે કે નાપાસ (જો કુલ ગુણ 35થી ઓછા હોય તો વિદ્યાર્થીને નાપાસ જાહેર કરો).
9. કોઈ વિદ્યાર્થીના કુલ ગુણને આધારે વિદ્યાર્થીએ કઈ શ્રેણી મેળવી છે તે શોધો. (જો વિદ્યાર્થીના કુલ ગુણ 35થી ઓછા હોય તો વિદ્યાર્થીને D શ્રેણી મળે છે. જો કુલ ગુણ 35 અને 60 વચ્ચે હોય તો C શ્રેણી અને જો કુલ ગુણ 60 અને 70 વચ્ચે હોય તો B શ્રેણી અને 70 કરતાં વધારે ગુણ હોય તો A શ્રેણી મેળવે છે.)
10. X ₹ મુદ્દલ રકમની લોન, R% વ્યાજનો દર અને N વર્ષના સમયગાળાનાં ચક્રવૃદ્ધિ વ્યાજની ગણતરી કરો.
11. આપેલી સંખ્યાના અંકોનો સરવાળો કરો. ઉદાહરણ તરીકે, જો આપેલી સંખ્યા 8327 હોય તો આઉટપુટ (8 + 3 + 2 + 7) = 20 થશે.