

അലൻ ട്യൂറിങ്

സന്തോഷ് തോട്ടിങ്ങൽ

Can Digital Computers Think?

Digital computers have often been described as mechanical brains. Most scientists probably regard this description as a mere newspaper stunt, but some do not. One well known mathematician has expressed the opposite point of view to me rather forcefully in the words 'It is commonly said that these machines are not brains, but you and I know that they are!'

ഇരുപതാം നൂറ്റാണ്ടിലെ വിപ്ലവകരമായ ഒരു കണ്ടുപിടിത്തമായിരുന്നു കമ്പ്യൂട്ടറുകളെന്ന് തർക്കമില്ലാത്ത വസ്തുതയാണ്. ജീവിതത്തിന്റെ സമസ്തമേഖലകളിലെയും സ്പർശിച്ചു കൊണ്ട് അതിവേഗത്തിലുള്ള നവീകരണത്തിന്റെ പാതയിലാണ് കമ്പ്യൂട്ടർ സയൻസ്. മനുഷ്യമനസ്സിനെ ഒരു യന്ത്രത്തിലേയ്ക്ക് പകർത്താമോ, അതിനെ മനുഷ്യബുദ്ധിക്ക് സമാനമായ കഴിവുള്ളതാക്കി മാറ്റാമോ എന്ന അന്വേഷണത്തിന്റെ ആദ്യത്തെ വിജയകരമായ ഉത്തരങ്ങളാണ് കമ്പ്യൂട്ടറുകളായി നമ്മുടെ ജീവിതത്തിൽ ഇടം പിടിച്ചിരിക്കുന്നത്. ഈ അന്വേഷണത്തിന്റെ ചരിത്രം അന്വേഷിക്കുന്ന ആർക്കും അലൻ ട്യൂറിങ് അസാമാന്യ ശാസ്ത്രപ്രതിഭയെ മറക്കാനാവില്ല. പ്രസിദ്ധ ഗണിതശാസ്ത്രജ്ഞൻ, കമ്പ്യൂട്ടർസയൻസ്, ആർട്ടിഫിഷ്യൽ ഇന്റലിജൻസ് എന്നിവയുടെ പിതാവ്, രണ്ടാംലോക മഹായുദ്ധകാലത്തെ രഹസ്യസന്ദേശങ്ങളെ മനസ്സിലാക്കിയെടുത്ത് ചരിത്രത്തെ സ്വാധീനിച്ച വ്യക്തി എന്നിങ്ങനെ പല മേഖലകളിൽ ഒരേസമയം കഴിവ് തെളിയിക്കുകയും അതേ സമയം വിധിയുടെ ക്രൂരതയിൽ നാൽപ്പത്തൊന്നാം വയസ്സിൽ ജീവനൊടുക്കേണ്ടിയും വന്ന പ്രതിഭ. അദ്ദേഹത്തിന്റെ ജന്മശതാബ്ധിയായ 2012 ട്യൂറിങ് വർഷം എന്ന പേരിൽ ആചരിയ്ക്കുന്നു.



ചിത്രം 1: അലൻ ട്യൂറിങ്

1912 ജൂൺ 23 നാണ് അലൻ ട്യൂറിങ് ജനിക്കുന്നത്. ബ്രിട്ടീഷ് ഇന്ത്യയിൽ മദ്രാസ് റെയിൽവെയിൽ ചീഫ് എൻജിനീയറായിരുന്ന ജൂലിയസ് മാത്തിസന്റെയും ഈതൽ സാറാ ട്യൂറിങ്ങിന്റെയും രണ്ടാമത്തെയും അവസാനത്തെയും പുത്രനായിരുന്നു ട്യൂറിങ്. രക്ഷിതാക്കളുടെ ജോലി ഒറിസയിലെ ചിത്രാപുരിലായിരുന്നെങ്കിലും ട്യൂറിങ് ജനിച്ചത് ലണ്ടനിൽ വെച്ചാണ്. ചെറുപ്പത്തിലെ ശാസ്ത്രത്തോടു ആഭിമുഖ്യം പുലർത്തിയിരുന്നെങ്കിലും പഠിച്ചിരുന്ന ഇംഗ്ലീഷ് പബ്ലിക് സ്കൂൾ അങ്ങനെയുള്ളവർക്ക് പറ്റിയതായിരുന്നില്ല. സയൻസിലാണ് താത്പര്യമെങ്കിൽ വേറെ സ്കൂൾ നോക്കാൻ ട്യൂറിങ്ങിന്റെ അമ്മയോടു്

ഹെഡ് മാസ്റ്റർ ഉപദേശിക്കുകയുണ്ടായി. ഗണിതശാസ്ത്രത്തിലായിരുന്നു ട്യൂറിങ്ങിനു താത്പര്യം. ഇരുപത്തിരണ്ടാം വയസ്സിൽ കിങ്സ് കോളേജിൽ നിന്ന് സാധ്യതാ സിദ്ധാന്തത്തിനോടു ബന്ധപ്പെട്ട് ട്യൂറിങ്ങിന് ഫെല്ലോഷിപ്പ് കിട്ടിയിരുന്നു. കണക്കിലും അദ്ദേഹത്തിന്റെ താത്പര്യം അവിടെ നിന്നില്ല.

ട്യൂറിങ് മെഷീൻ

കമ്പ്യൂട്ടറിന്റെ മുന്നിൽ നിങ്ങളിരിക്കുമ്പോൾ മിക്കവാറും നിങ്ങളറിഞ്ഞിട്ടുണ്ടാവില്ല അതൊരു ട്യൂറിങ് മെഷീനാണെന്ന്! എന്താണ് ട്യൂറിങ് മെഷീൻ എന്നറിയണമെങ്കിൽ നമുക്ക് കുറച്ച് ചരിത്രം അറിയേണ്ടിയിരിക്കുന്നു. ഗണിതശാസ്ത്ര പ്രശ്നങ്ങളിലൂടെ മനുഷ്യൻ യാന്ത്രിക കണക്കുകൂട്ടലുകൾ നടത്താനുള്ള വഴി കണ്ടുപിടിച്ച ചരിത്രം.

പത്തൊമ്പതാം നൂറ്റാണ്ടിന്റെ അവസാനത്തിലും ഇരുപതാം നൂറ്റാണ്ടിന്റെ ആദ്യ ദശകങ്ങളിലും ഗണിത ശാസ്ത്രത്തിന്റെ കെട്ടുറപ്പിനെപ്പറ്റിയുള്ള നിരവധി ചർച്ചകൾ നടക്കുകയുണ്ടായി. കണക്കിലെ സിദ്ധാന്തങ്ങൾ തെളിയിക്കാൻ പല ആക്സിയങ്ങൾ (Axiom) അഥവാ മൗലികതാങ്ങൾ ഉപയോഗിക്കുന്നത് എല്ലാവർക്കും അറിയുന്നതാണല്ലോ. ഇല്ലെങ്കിൽ ലളിതമായി പറയാം. ഒരു സിദ്ധാന്തം തെളിയിക്കാൻ അതിനു മുന്നേ തെളിയിച്ച പല സിദ്ധാന്തങ്ങളെയും നമ്മൾ കൂട്ടുപിടിക്കും. പക്ഷേ ആ കൂട്ടുപിടിച്ച സിദ്ധാന്തങ്ങളും തെളിയിക്കപ്പെട്ടിട്ടുള്ളത് വേറെ പലതിന്റെയും അടിസ്ഥാനത്തിലാവും. ഇങ്ങനെ നമ്മൾ പുറകോട്ട് പോയാൽ ഒരു കൂട്ടം അടിസ്ഥാനപ്രമാണങ്ങൾ തെളിവുകളൊന്നുമില്ലാതെ, സാമാന്യബുദ്ധിയുടെ അടിസ്ഥാനത്തിൽ ഗണിത ശാസ്ത്രത്തിലുള്ളതായിക്കാണാം. ഉദാഹരണത്തിന് "ഒരു വസ്തുവിന് തുല്യമായ വസ്തുക്കളെല്ലാം പരസ്പരം തുല്യങ്ങളായിരിക്കും", "മുഴുവനെക്കാൾ ചെറുതാണ് ഭാഗികം" തുടങ്ങിയവ. ഈ അടിസ്ഥാന പ്രമാണങ്ങളുടെ ഉറപ്പിലാണ് ഗണിതത്തിൽ സങ്കീർണ്ണമായ സിദ്ധാന്തങ്ങൾ പടുത്തുയർത്തുന്നത്. ഇവയിലൊന്നു തെറ്റാണെങ്കിലോ? ഗണിത ശാസ്ത്രത്തിനു വരുന്ന വലിയ വിപത്താവില്ലേ അത്? സത്യത്തിൽ അത്തരം ചില പ്രശ്നങ്ങൾ കണക്കിൽ ഉണ്ടായിരുന്നു. ഉദാഹരണത്തിന് റസ്സൽ വിരോധാഭാസം (Russel's Paradox) #. അതുകൊണ്ട് തെറ്റില്ലാത്തതെന്ന് നൂറുശതമാനവും ഉറപ്പിച്ചുപറയാവുന്ന ഒരു കൂട്ടം സിദ്ധാന്തങ്ങൾ ഉണ്ടാക്കാൻ ഡേവിഡ് ഹിൽബർട്ട് എന്ന പ്രശസ്ത ജർമ്മൻ ഗണിത ശാസ്ത്രജ്ഞൻ ശ്രമം തുടങ്ങിയിരുന്നു.

റസ്സൽ വിരോധാഭാസം (Russel's Paradox)

സ്വയം അംഗമല്ലാത്ത എല്ലാ ഗണങ്ങളുടെയും ഒരു ഗണമാണ് R എന്നിരിക്കട്ടെ. R എന്ന ഈ ഗണത്തിൽ R ഒരംഗമാവുമോ? പറ്റില്ല - കാരണം R ൽ R സ്വയം അംഗമാണെങ്കിൽ അത് R ന്റെ നിർവചനത്തിനെതിരാണ്. അപ്പോൾ R ൽ R ഇല്ല. പക്ഷേ അതെങ്ങനെ ശരിയാവും. സ്വയം അംഗമില്ലാത്ത എല്ലാ ഗണങ്ങളും R ൽ ഉണ്ടെങ്കിൽ, R തീർച്ചയായും R ൽ വേണ്ടേ? ഈ വിരോധാഭാസമാണ് റസ്സൽ വിരോധാഭാസം. അടിസ്ഥാനസിദ്ധാന്തങ്ങൾ പലതും ഗണസിദ്ധാന്തമുപയോഗിച്ച് തെളിയിച്ചിരുന്ന കാലത്ത് ഗണസിദ്ധാന്തത്തിലെ ഈ വിരോധാഭാസം ഗണിതശാസ്ത്രത്തിന്റെ തലവേദനയായിരുന്നു. ഇങ്ങനെ ഒരു പ്രശ്നം ഉണ്ടായത് അടിത്തറയിൽ എവിടെയോ പിഴച്ചുകൊണ്ടല്ലേ എന്ന ചോദ്യം ഉയർന്നിരുന്നു.

സമാനവും രസകരവുമായ വേറൊന്ന്: ഗ്രാമത്തിലെ സ്വയം താടിവടിക്കാത്ത ആണങ്ങളുടെയെല്ലാം താടിവടിക്കുന്നത് ക്ഷുരകനാണ്. അപ്പോൾ ക്ഷുരകന്റെ താടി ആരു വടിക്കും? ക്ഷുരകൻ തന്നെയാണെങ്കിൽ ആദ്യം പറഞ്ഞത് തെറ്റി- സ്വയം താടിവടിക്കാത്തവരെ മാത്രമേ ക്ഷുരകൻ വടിക്കുന്നുള്ളൂ. വേറൊരാളാണെങ്കിലോ? ആ ഗ്രാമത്തെ സംബന്ധിച്ചിടത്തോളം അത് ക്ഷുരകൻ തന്നെയാണ്. ഇത് ക്ഷുരക വിരോധാഭാസം (Barber Paradox) എന്നറിയപ്പെടുന്നു.



ചിത്രം 2: ഡേവിഡ് ഹിൽബർട്ട്.

പ്രശ്നനിർദ്ധാരണത്തിനും സിദ്ധാന്തങ്ങൾ തെളിയിക്കുന്നതിനും കൃത്യമായ ഒരു മാർഗ്ഗരേഖയുണ്ടാക്കലും ഹിൽബർട്ടിന്റെ ലക്ഷ്യമായിരുന്നു. സിദ്ധാന്തങ്ങളെക്കുറിച്ച് വിശകലനം നടത്തുമ്പോഴും തെളിയിക്കുമ്പോഴും ഉണ്ടാവുന്ന മനുഷ്യസഹജമായ തെറ്റുകൾ ഒഴിവാക്കാൻ തക്കവണ്ണം എങ്ങനെ ഒരു ഗണിതക്രിയയെ ലളിതവും കൃത്യവുമാക്കാം എന്നായിരുന്നു അദ്ദേഹത്തിന്റെ ചിന്ത. ഇതിനായി ഹിൽബർട്ട് 23 ചോദ്യങ്ങൾ ആവിഷ്കരിച്ചു. കെട്ടുറപ്പുള്ള ആക്ഷീയങ്ങളെ ലക്ഷ്യമിട്ട ഹിൽബർട്ടിന്റെ പദ്ധതിയ്ക്ക് വൻ ആഘാതമായി 1931 ൽ കർട്ട് ഗോദൽ(Kurt Friedrich Gödel) എന്ന 25 വയസ്സുണ്ടായിരുന്ന ആസ്ട്രേലിയൻ ഗണിതശാസ്ത്രജ്ഞൻ ഗോദൽ അപൂർണ്ണതാ സിദ്ധാന്തം (Incompleteness theorem) അവതരിപ്പിച്ചു. ഇതനുസരിച്ച് ആക്ഷീയങ്ങളെ അടിസ്ഥാനമാക്കിയുള്ള ഒരു ഗണിത ശാസ്ത്ര സംവിധാനം(axiomatic system) ഒരിക്കലും നിലനിൽക്കുന്നതല്ല. അതിലെ ഒരു ആക്ഷീയമെങ്കിലും ആ സംവിധാനത്തിനു പുറത്ത് തെളിയിക്കേണ്ടിവരും. അതായത് സംവിധാനത്തിനകത്തെ ആക്ഷീയങ്ങളെല്ലാം ശരിയെന്നിരിക്കിലും അല്ലെങ്കിൽ ശരിയെന്നു തെളിയിക്കാമെങ്കിലും(Consistent system) അങ്ങനെ തെളിയിക്കണമെങ്കിൽ ബാഹ്യമായ ഒരു ആക്ഷീയം അതിനാവശ്യമാവും. ഹിൽബർട്ടിന്റെ ചോദ്യാവലിയിലെ രണ്ടാമത്തെ ചോദ്യം ഇതായിരുന്നു: “അങ്കഗണിതത്തിലെ (Arithmetics) ആക്ഷീയങ്ങളെല്ലാം കെട്ടുറപ്പുള്ളതാണെന്നു തെളിയിക്കാമോ?” (Prove that the axioms of arithmetic are consistent.). അതുപറ്റില്ലെന്നു തെളിയിക്കപ്പെട്ടു!. ആധുനിക ഗണിതത്തിലെ നിലപാട് ആക്ഷീയങ്ങൾ പരിപൂർണ്ണമായും സത്യമാണെന്നല്ല, മറിച്ച് അവ സത്യമെന്നു സ്വീകരിക്കപ്പെട്ടവയാണെന്നും അവ സത്യമാകുന്നിടത്തെല്ലാം അവയിൽ നിന്നു സിദ്ധിച്ച പ്രമേയങ്ങളും സത്യമായിരിക്കും എന്നും മാത്രമാണ്.

ഹിൽബർട്ടിന്റെ പദ്ധതിയെപ്പറ്റിയും ഗോദലിന്റെ സിദ്ധാന്തങ്ങളെപ്പറ്റിയും ട്യൂറിങ്ങിനറിയാമായിരുന്നു. 1935 ൽ ഹിൽബർട്ടിന്റെ നിർണ്ണയതാ ചോദ്യത്തിന്(Decidability) ആരും ഉത്തരം കണ്ടെത്തിയിട്ടില്ല എന്നു ട്യൂറിങ് മനസ്സിലാക്കി. ഒരു ഗണിതശാസ്ത്ര നിഗമനം ശരിയെന്നു തെളിയിക്കാൻ കൃത്യമായ ഒരു രീതിയോ പ്രക്രിയയോ തത്വത്തിലേകിലും നിലനിൽക്കുന്നുണ്ടോ എന്നതായിരുന്നു ആ ചോദ്യം. ഇതുകൊണ്ട് ഹിൽബർട്ട് ഉദ്ദേശിച്ചത് ഒരു ഗണിതശാസ്ത്ര പ്രക്രിയയെ ഒരു യാന്ത്രിക പ്രക്രിയയാക്കി മാറ്റാൻ മാർഗ്ഗങ്ങളെന്തെങ്കിലുമുണ്ടോ എന്നാണെന്നു മനസ്സിലാക്കാം. യാന്ത്രിക പ്രക്രിയ എന്നാൽ മനുഷ്യന്റെ ബുദ്ധിയുടെ ഇടപെടലില്ലാതെ കൃത്യമായി നിർവചിച്ചിരിക്കുന്ന പടിപടിയായ പ്രക്രിയകളിലൂടെ ഗണിതക്രിയകളോ നിഗമനങ്ങളോ സാധ്യമാവുമോ എന്നാണ്.

അങ്ങനെയുള്ള ഒരു യാന്ത്രിക പ്രക്രിയയെ എങ്ങനെ നിർവചിക്കും? ഇതാണ് ആ നിർവചനം:

1. നിർവചിക്കാവുന്ന എണ്ണം ചിഹ്നങ്ങൾ അടങ്ങിയ നിർവചിക്കാവുന്ന എണ്ണം നിർദ്ദേശങ്ങളുടെ കൂട്ടമാണ് ആ പ്രക്രിയ.
2. ഈ നിർദ്ദേശങ്ങൾ തെറ്റുകൂടാതെ ഒരു നിശ്ചിത എണ്ണം പടിപടിയായി നടപ്പിലാക്കിയാൽ ഉദ്ദേശിച്ച ഫലത്തിൽ നാം എത്തിച്ചേരും
3. ഈ നിർദ്ദേശങ്ങൾ ഒരു മനുഷ്യൻ ഒരു യന്ത്രത്തിന്റെയും സഹായമില്ലാതെ പേപ്പറും പെൻസിലുമുപയോഗിച്ച് ചെയ്യാൻ കഴിയും.തത്വത്തിൽ മാത്രമല്ല, പ്രായോഗികമായും ചെയ്യാൻ കഴിയണം.
4. ഈ നിർദ്ദേശങ്ങളുടെ നടത്തിപ്പിന് നടപ്പിലാക്കുന്ന ആളുടെ ബുദ്ധിയോടോ കഴിവിനോ-

അലൻ ടൂറിങ്ങ്

ടോ അനുഭവപരിചയമായോ ഒരു ബന്ധവുമില്ല.

1920 നോടകം തന്നെ ഈ നിർവചനം(Effective Methods) ഗണിതലോകത്തിന് പരിചിതമായിരുന്നു. കമ്പ്യൂട്ടറുകൾ പ്രയോഗത്തിൽ വരുന്നതിനു ഏറെ മുമ്പ്! ഇന്ന് അൽഗോരിതം എന്നറിയപ്പെടുന്ന കമ്പ്യൂട്ടർ പ്രോഗ്രാമിന്റെ നടപടിക്രമം തന്നെയല്ലേ ഇത്? ഈ പ്രക്രിയ നമുക്ക് അത്ര അപരിചിതമൊന്നുമല്ല. വലിയ സംഖ്യകളെ ഗുണിപ്പാനും ഹരിക്കാനും, ഉസാഹ, ലസാഗു എന്നിവ കാണാനും ഒക്കെ ചെറിയ ക്ലാസ്സുകളിൽ നാം ശീലിച്ച "വഴികൾ" ആണിത്. വഴിയെഴുതി ക്രിയ ചെയ്യുക എന്ന കണക്കുചോദ്യങ്ങൾ ഓർമ്മയില്ലേ? ഈ നിർവചനത്തിലെ മൂന്നാം ഭാഗം നോക്കുക. അതനുസരിച്ച് ഈ പ്രക്രിയ ഒരു മനുഷ്യൻ ചെയ്യാനുള്ളതാണ്. തെറ്റുകളും അബദ്ധങ്ങളുമില്ലാതെ ആർക്കും കൃത്യമായി ഉത്തരത്തിലെത്തിച്ചേരാനുള്ള പ്രക്രിയ- നാലാം ഭാഗമനുസരിച്ച് ബുദ്ധിയുടെയും പരിചയത്തിനെയും ഒന്നും ആശ്രയിക്കാതെ. എന്നുവെച്ചാൽ അവനൊരു വെറും ജൈവ യന്ത്രമല്ലേ? ആ മനുഷ്യൻ ചെയ്യുന്നത് ഒരു യന്ത്രത്തിനെ കൊണ്ട് ചെയ്യിക്കാനാവുമോ? ടൂറിങ്ങ് ഈ പ്രശ്നത്തിനത്തരം കണ്ടെത്താൻ ശ്രമിച്ചു.

അതിനായി ടൂറിങ്ങ് ആദ്യം ചിന്തിച്ചത് മനുഷ്യൻ എങ്ങനെ ഒരു ഗണിതക്രിയ ചെയ്യുന്നു എന്നാണ്. പേപ്പറും പെൻസിലും വേണമെന്നു പറഞ്ഞപ്പോൾ, ഈ പേപ്പർ പുസ്തകത്തിന്റെ പേജ് പോലിരിക്കണോ, അതോ ഒരു നാടപോലെ ഇരുന്നാൽ മതിയോ? ടൂറിങ്ങ് ആദ്യം അതിനെ ഒരു നാടയായി സങ്കല്പിച്ചു. ഇടത്തോട്ടും വലത്തോട്ടും എഴുതാവുന്ന നാട. ഇതിനെത്ര നീളം വേണ്ടിവരും? നിർദ്ദേശങ്ങളെല്ലാം ചെയ്ത് തീരും വരെ പേപ്പർ തീരാൻ പാടില്ല. സൗകര്യത്തിനായി ഈ നാടയ്ക്ക് അനന്തമായ നീളമുണ്ടെന്നു കരുതാം- ഇടത്തോട്ടും വലത്തോട്ടും. പെൻസിൽ ഒരിക്കലും തേഞ്ഞു തീരാത്തതും!. ആവശ്യാനുസരണം പേപ്പറിലെഴുതിയത് മായ്ക്കുകയും ചെയ്യാം. വേണ്ടാത്ത സങ്കീർണ്ണതകൾ ഒഴിവാക്കാൻ ഈ മനുഷ്യൻ വിശപ്പും ദാഹവുമില്ല, ആരും ശല്യപ്പെടുത്തുന്നില്ല. മടിയെന്തെന്ന് ഇവനിയില്ല.

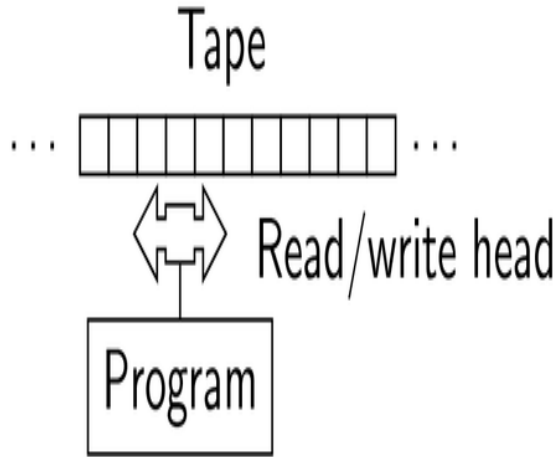
സംഖ്യ	ബൈനറിയിൽ		
0			0
1			1
2		1	0
3		1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1

വളരെ ലളിതമായി ഒന്ന്, രണ്ട്, മൂന്ന് എന്നിങ്ങനെ സംഖ്യകൾ തുടർച്ചയായി എഴുതുന്നതാണു നമ്മുടെ പ്രക്രിയ എന്നിരിക്കട്ടെ. അതെങ്ങനെയിരിക്കും? നമുക്കു നോക്കാം. സൗകര്യാർത്ഥം ബൈനറിയിലാണ് - ഒന്നും പൂജ്യവും ഉപയോഗിച്ചുള്ള സംഖ്യാസമ്പ്രദായത്തിലാണ് നമ്മൾ എഴുതുന്നത്. ടേബിൾ നോക്കുക. പേപ്പറിലൊന്നുമില്ലെങ്കിൽ നമുക്കതിനെ 000 എന്നു പറയാം. ഇതു പൂജ്യമാണ്. ഇതിന്റെ കൂടെ 1 കൂട്ടണം. ബൈനറി അക്കങ്ങൾ നോക്കിയാൽ മനസ്സിലാവും 000 എന്നതിലെ അവസാനത്തെ 0 മായിച്ചുകളഞ്ഞ് 1 എന്നെഴുതിയാൽ മതി. 2, അതായത് 10, അവസാനത്തെ 1 മായിച്ചുകളഞ്ഞ് 0 എഴുതുകയും നാട വലത്തോട്ട് നീക്കി 1 എന്നെഴുതുകയും വേണം. ഈ പ്രക്രിയയെ സ്വാധീനിക്കുന്ന ഘടകങ്ങൾ എന്തൊക്കെയാണ്?

1. അവസ്ഥകൾ- ഒരു തുടക്കം, ഒരു അവസാനം, അതിനിടയ്ക്കുള്ള അവസ്ഥകൾ. എവിടെ എത്തി എന്നതുതന്നെ.
2. നാടയിൽ എന്തെങ്കിലും എഴുതിട്ടുണ്ടോ? എന്താണത്
3. നാടയിൽ എന്തെങ്കിലും എഴുതാനുണ്ടോ? ഉണ്ടെങ്കിലിത്. അത് നാടയിൽ ഇപ്പോൾ ഉള്ളതു തന്നെയാണെങ്കിൽ ഒന്നും ചെയ്യേണ്ട.
4. അടുത്ത ക്രിയക്കു മുമ്പ് നാട ഇടത്തോട്ടോ വലത്തോട്ടോ നീക്കണോ? അതോ നിന്നിടത്തു തന്നെ നിൽക്കണോ?
5. ക്രിയ ചെയ്തശേഷം(എന്നുവെച്ചാൽ നാടയിൽ എഴുതാനുണ്ടെങ്കിൽ എഴുതിയശേഷം) ഏതവസ്ഥയിലേക്ക് പോകണം.

പേപ്പറും പെൻസിലും ഒക്കെ കളഞ്ഞ് ഈ പ്രക്രിയകൾ ഒരു യന്ത്രത്തെക്കൊണ്ടു ചെയ്യിച്ചാൽ അതൊരു "എണ്ണൽ" യന്ത്രമായില്ലേ? മേൽപ്പറഞ്ഞതിൽ മനുഷ്യനുമാത്രം ചെയ്യാൻ പറ്റും യന്ത്രത്തിനു ചെയ്യാൻ കഴിയില്ല എന്ന രീതിയിൽ

എന്തെങ്കിലുമുണ്ടോ? ഇല്ല. അപ്പോൾ ഇനി ഈ യന്ത്രത്തിലേതൊക്കെ വേണം? നാട. വളരെ നീണ്ട നാട. അതു നമുക്ക് ഒരു മോട്ടോറിൽ ഇട്ട് ഇടത്തോട്ടും വലത്തോട്ടും നീക്കാം. നാടയിൽ എഴുതിയത് വായിച്ച്, എഴുതാൻ കഴിയുന്ന ഒരു ഉപകരണം. Read-Write Head എന്നുപറയാം. ഇനിയെങ്കിലും ഇടത്തോട്ട് നീങ്ങണോ, വലത്തോട്ട് നീങ്ങണോ, എഴുതേണ്ടത് എന്നൊക്കെ എഴുതിയ ഒരു പട്ടികയും. ഇതാണ് ട്യൂറിങ്ങ് മെഷീൻ!



ചിത്രം 3: ട്യൂറിങ്ങ് മെഷീൻ.

ഇടക്കത്തിൽ പറഞ്ഞു നമ്മൾ ഉപയോഗിക്കുന്ന കമ്പ്യൂട്ടറുകളൊക്കെ ട്യൂറിങ്ങ് മെഷീനുകളാണെന്ന്. സങ്കീർണ്ണങ്ങളായ കാര്യങ്ങൾ ചെയ്യുന്ന കമ്പ്യൂട്ടററിന്റെ ഈ നാടയും റീഡ്/റൈറ്റ് ഹെഡും എവിടെക്കിടക്കുന്നുവെന്ന് നിങ്ങൾക്കു തോന്നും. അതിലേയ്ക്കു നമ്മൾ ഉടനെ എത്താം. നമ്മുടെ "എണ്ണൽ" പ്രക്രിയയെ യന്ത്രത്തിലേക്കാൻ ശ്രമിക്കാം. നാടയുടെയും റീഡ്/റൈറ്റ് ഹെഡിന്റെയും കാര്യം പറഞ്ഞു. പക്ഷേ നിർദ്ദേശങ്ങളടങ്ങിയ പട്ടികയുടെ കാര്യം പറഞ്ഞിട്ടില്ല.

അവസ്ഥ	വായിച്ചത്	പോകേണ്ട അവസ്ഥ	എഴുതേണ്ടത്	ഹെഡ് നീങ്ങേണ്ടത്	വിശദീകരണം
0	1	0	1	വലത്തോട്ട്	ഒന്നാം മാറ്റിയെഴുതാതെ 0 അവസ്ഥയിൽ തന്നെ വലത്തോട്ട് നീങ്ങുക.
0	0	0	0	വലത്തോട്ട്	ഒന്നാം മാറ്റിയെഴുതാതെ 0 അവസ്ഥയിൽ തന്നെ വലത്തോട്ട് നീങ്ങുക.
0	ശൂന്യം	1	ശൂന്യം	ഇടത്തോട്ട്	ഒന്നാം വായിച്ചില്ല. പേപ്പറിലെ അക്കത്തിന്റെ അവസാനത്തിലാണിത്. ഇടത്തോട്ട് നീങ്ങുക, അവസ്ഥ 1 ലേയ്ക്ക് പോവുക.
1	0	0	1	വലത്തോട്ട്	0 കണ്ടാൽ 1 ആക്കുക. അവസ്ഥ 0 ൽ വലത്തോട്ട് നീങ്ങുക.
1	1	1	0	ഇടത്തോട്ട്	1 കണ്ടാൽ 0 ആക്കുക. അവസ്ഥ 1 ൽ തന്നെ ഇടത്തോട്ട് നീങ്ങുക.
1	ശൂന്യം	0	1	വലത്തോട്ട്	ഒന്നാം കണ്ടില്ലെങ്കിൽ 1 ആക്കുക. അവസ്ഥ 0 ൽ വലത്തോട്ട് നീങ്ങുക.

ഈ നിയമങ്ങൾ ഉപയോഗിച്ച് എണ്ണുന്നതിന്റെ ഒരു വീഡിയോ ദൃശ്യം <http://ur1.ca/9gkit> എന്ന വിലാസത്തിലുണ്ട്.

ജീജ്ഞാസുക്കളായ വായനക്കാർക്ക് മുകളിലെ നിർദ്ദേശങ്ങൾ പ്രകാരം ഏത് സംഖ്യ തന്നാലും അതിന്റെ തൊട്ടടുത്ത സംഖ്യ കാണാൻ കഴിയുമോയെന്ന് പരീക്ഷിച്ചു നോക്കാവുന്നതാണ്. ഇതുപോലെ കുറയ്ക്കൽ, ഹരിയ്ക്കൽ തുടങ്ങി ഏതു ഗണിതക്രിയയ്ക്കും തുല്യമായ ട്യൂറിങ്ങ് മെഷീനുണ്ടാക്കാം. കാര്യമൊക്കെ ശരി. കൂട്ടാനുള്ള ട്യൂറിങ്ങ് മെഷീനുണ്ടാക്കാം. കുറയ്ക്കാനുള്ളതുണ്ടാക്കാം. ഇവയൊക്കെ വെവ്വേറെ ട്യൂറിങ്ങ് മെഷീനുകളാവില്ലേ? നാം ഇന്നു കാണുന്ന കമ്പ്യൂട്ടറിൽ ഏതു ഗണിതക്രിയയും ചെയ്യാമല്ലോ? ഒരു ക്രിയയെപ്പോലും ഒരു മെഷീൻ എന്നതല്ലല്ലോ ഇന്നത്തെ രീതി? ഈ ചോദ്യത്തിനത്തരം കാണാം.

ഒരു ട്യൂറിങ്ങ് മെഷീൻ എന്ന് ഒരു നിർദ്ദേശപ്പട്ടിക മാത്രം അനുസരിയുന്നതാണല്ലോ. ആ നിർദ്ദേശങ്ങൾ കറേ അക്കങ്ങളും അക്ഷരങ്ങളുമൊക്കെ ഉള്ള പദാവലികളാണ്. പക്ഷേ ഈ നിർദ്ദേശപ്പട്ടിക മൊത്തത്തിൽ ഒരു പദാവലിയിലേയ്ക്കോ അക്കത്തിലേയ്ക്കോ നമുക്ക് എൻകോഡ് ചെയ്യാം. വളരെലളിതമായിപ്പറഞ്ഞാൽ നമ്മുടെ എണ്ണൽ യന്ത്രത്തിന്റെ പട്ടികയിലെ അക്കങ്ങളും അക്ഷരങ്ങളെയും ഒക്കെ പട്ടികയും കള്ളിയുമൊന്നുമില്ലാതെ നീട്ടിയങ്ങുതേങ്ങി ഒരു വാക്കാക്കിക്കൂടെ? ഇങ്ങനെ ഒരു വാക്കിലേയ്ക്കോ അല്ലെങ്കിൽ ഒരു അക്കത്തിലേയ്ക്കോ എൻകോഡ് ചെയ്യപ്പെട്ട വലിയൊരു ട്യൂറിങ്ങ് മെഷീനുകളുടെ ശ്രേണി വായിച്ച് അതിലെ ഓരോ ട്യൂറിങ്ങ് മെഷീനുകളെയും സിമുലേറ്റ് ചെയ്ത് പ്രവർത്തിപ്പിക്കുന്ന ഒരു വമ്പൻ ട്യൂറിങ്ങ് മെഷീൻ സാദ്ധ്യമാണെന്ന് ട്യൂറിങ്ങ് വാദിച്ചു. ഇതൊരു സിദ്ധാന്തമാണ്, ഗണിതശാസ്ത്രപ്രകാരം തെളിയിക്കാവുന്നതു്. പക്ഷേ ഇതിന്റെ യന്ത്രരൂപത്തിലുള്ള ഒരു നിർമ്മിതി നമ്മുടെ എണ്ണൽ യന്ത്രത്തെക്കാൾ വളരെ സങ്കീർണ്ണവും സമയമെടുക്കുന്നതുമാണ്. പക്ഷേ അസാദ്ധ്യമല്ല.

ഈ വമ്പൻ ട്യൂറിങ്ങ് മെഷീനെ ട്യൂറിങ്ങ് വിളിച്ചത് യൂണിവേഴ്സൽ ട്യൂറിങ്ങ് മെഷീൻ എന്നായിരുന്നു. ട്യൂറിങ്ങ് ഈ ഗവേഷണങ്ങൾ നടത്തുമ്പോൾ മെഷീൻ എന്ന രൂപകമുപയോഗിക്കാതെ ഗണിത ശാസ്ത്രത്തിൽ സമാനമായ ഗവേഷണം നടത്തുന്ന വേറൊരു ശാസ്ത്രജ്ഞനുമുണ്ടായിരുന്നു- അലോൻസോ ചർച്ച്. അദ്ദേഹം ലാംഡാ കാൽക്കുലസ് ഉപയോഗിച്ച് ഏതൊരു കണക്കുകളും നടത്താവുന്ന ഒരു ഗണിതശാസ്ത്ര രീതി നിർമ്മിച്ചെടുക്കാനുള്ള ശ്രമത്തിലായിരുന്നു. അദ്ദേഹത്തിന്റെ ലക്ഷ്യം ഗണിതശാസ്ത്രത്തിൽ Effective Methods (ലേഖനത്തിന്റെ ആദ്യഭാഗത്ത് വിശദീകരിച്ചിട്ടുണ്ട്) കൊണ്ടുവരികയായിരുന്നു. തെറ്റില്ലാത്ത പ്രശ്നനിർദ്ധാരണത്തിനദ്ദേഹം കൂട്ടുപിടിച്ചത് ലാംഡാ കാൽക്കുലസ് ആണെന്നു മാത്രം. ഇതിനിടയ്ക്ക് നമ്മൾ ഒരു ട്യൂറിങ്ങ് മെഷീനെ ഒരു വാക്കിലോ അക്കത്തിലോ എൻകോഡ് ചെയ്യാമെന്നു പറഞ്ഞല്ലോ. ഇത് ശരിക്കും ചെയ്യുന്നത് ഗൊദൽ നമ്പറിങ്ങ് എന്ന സൂത്രമുപയോഗിച്ചാണ്. കൂർട്ട് ഗൊദൽ ഇതുകണ്ടുപിടിച്ചത് ഗണിതപ്രശ്ന നിർദ്ധാരണത്തിലെ പടികളെ(steps) ഗണിതശാസ്ത്രം കൊണ്ടുതന്നെ കുറ്റമറ്റതാക്കാൻ പറ്റുമോ എന്നു നോക്കാനായിരുന്നു. അതായത് $a = x + y$ തുടങ്ങിയ ഗണിത പ്രസ്ഥാവനകളെ ഒരു നമ്പർ കൊണ്ട് പ്രതിനിധീകരിച്ചുകൊണ്ടുള്ള രീതി. ഈ രീതിയിൽ അദ്ദേഹം കൂട്ടുപിടിച്ചത് റിക്കുവ് അരിത്മെറ്റിക് എന്ന രീതിയാണ്. എഫക്ടീവ് മെത്തേഡ്സ് ഒരു ട്യൂറിങ്ങ് മെഷീൻ കൊണ്ട് സാധിച്ചെടുക്കാമെന്ന് നമ്മൾ കണ്ടു. എഫക്ടീവ് മെത്തേഡ്സിനെ ലാംഡാ കാൽക്കുലസ് ഉപയോഗിച്ച് നിർവചിക്കാമെന്നു ചർച്ചും കണ്ടുപിടിച്ചു. ഒരു ലാംഡാ കാൽക്കുലസ് നിർദ്ധാരണത്തിനുള്ള ട്യൂറിങ്ങ് മെഷീൻ സാദ്ധ്യമാണെന്ന് ട്യൂറിങ്ങ് കണ്ടുപിടിച്ചു. അങ്ങനെയെങ്കിൽ എഫക്ടീവ് മെത്തേഡുകൾ എല്ലാം ട്യൂറിങ്ങ് മെഷീൻ കൊണ്ട് സാദ്ധ്യമാണെന്നു വരുന്നു. ഇത് ചർച്ച്-ട്യൂറിങ്ങ് തീസിസ് എന്നറിയപ്പെടുന്നു. (ഈ സിദ്ധാന്തങ്ങളെല്ലാം തന്നെ രസകരമാണെങ്കിലും അവ വിശദീകരിക്കുന്നത് ഈയവസരത്തിൽ അനുചിതമാണെന്നതിനാൽ ജീജ്ഞാസുക്കളായ വായനക്കാർക്ക് ഇന്റർനെറ്റ് അല്ലെങ്കിൽ ഒരു ലൈബ്രറിയുടെ സഹായത്തോടെ ഇവ കൂടുതൽ മനസ്സിലാക്കാവുന്നതാണ്.)

ഏതു ട്യൂറിങ്ങ് മെഷീനെയും സിമുലേറ്റ് ചെയ്യാൻ കഴിയുന്ന ഒരു ട്യൂറിങ്ങ് മെഷീനാണ് യൂണിവേഴ്സൽ ട്യൂറിങ്ങ് മെഷീൻ എന്നു കണ്ടു. ഇത് കമ്പ്യൂട്ടറിനു ജന്മം കൊടുക്കാനുള്ള ബീജമായിരുന്നു. സ്റ്റാൻഫോർഡ് എൻസൈക്ലോപീഡിയ ഓഫ് കമ്പ്യൂട്ടിങ്ങ് ഇങ്ങനെ പറയുന്നു.

ട്യൂറിങ്ങിന്റെ യൂണിവേഴ്സൽ ട്യൂറിങ്ങ് മെഷീൻ കമ്പ്യൂട്ടേഷൻ എന്ന പ്രക്രിയയ്ക്ക് മൗലികമായ അടിത്തറ ഇടുന്നു - ഏതു പ്രോഗ്രാമും ഓടിക്കാവുന്ന ഒരു മെഷീൻ. ഭാവിയിലെന്തു കമ്പ്യൂട്ടേഷനും നമുക്ക് ചെയ്യണമെന്നിരിക്കട്ടെ, എല്ലാം ചെയ്യേണ്ടതാവുന്ന ഒരൊറ്റ മെഷീൻ. ഈ ഉൾക്കാഴ്ചയാണ് ഇന്ന് നമ്മെ കമ്പ്യൂട്ടറുകൾ നിർമ്മിക്കാനും വിൽക്കാനും

പ്രാപ്തമാക്കുന്നത്. ഏതു പ്രോഗ്രാമും ഒരു കമ്പ്യൂട്ടറിൽ ഓടും. ഓരോ പുതിയ പ്രശ്നത്തിനും ഓരോ കമ്പ്യൂട്ടറല്ല. തീർച്ചയായും ഇന്നത്തെ പേഴ്സണൽ കമ്പ്യൂട്ടർ യുഗത്തിൽ മുഴുകിയിരിക്കുന്ന നമുക്ക് ഒന്ന് പുറകോട്ട് നിന്നു ഈ വസ്തുതകൾ ആസ്വദിക്കാൻ ബുദ്ധിമുട്ടാണ്.

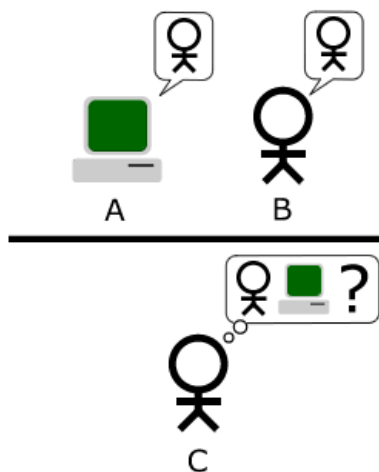
അതെ, ഹൈ ലെവൽ പ്രോഗ്രാമിങ്ങ് പഠിച്ചുകൊണ്ടുണ്ടുണ്ടുന്ന ഇന്നത്തെ കമ്പ്യൂട്ടർ വിദ്യാഭ്യാസരീതിയിൽ, അടിത്തറയിലേക്കും വന്ന വഴിയിലേയ്ക്കും നോക്കുന്നതും മനസ്സിലാക്കാൻ ശ്രമിക്കുന്നതും കുറേ ബുദ്ധിമുട്ടുള്ള കാര്യമാണ്. ആരെയും കുറ്റം പറഞ്ഞിട്ടുകാര്യമില്ല. യൂണിവേഴ്സിറ്റി ട്യൂറിങ്ങ് മെഷീന്റെ നിർവചനത്തോടുകൂടി കമ്പ്യൂട്ടറിന്റെ ചരിത്രവഴിയിൽ നിന്നും ഗണിതശാസ്ത്രം പതിയെ ഇലക്ട്രോണിക്സിലേയ്ക്ക് വഴിമാറിക്കൊടുത്തു. ശാസ്ത്രജ്ഞരുടെ കയ്യിൽ നിന്നും അത് എൻജിനീയർമാരുടെ കൈകളിലേക്ക് വഴിമാറി. അത്ര എളുപ്പമായിരുന്നില്ല കാര്യങ്ങൾ. ട്യൂറിങ്ങിന്റെ നാട ഇന്നത്തെ RAM, Harddisk എന്നീ രൂപത്തിലേത്താൻ കാലങ്ങൾ പിന്നെയും എടുത്തു. അനന്തമായ നാട ഇന്ന് അത്ര അനന്തമല്ലെങ്കിലും പ്രായോഗികമായ ഏതാവശ്യത്തിനും മതിയാകുന്ന രീതിയിലാണ് കമ്പ്യൂട്ടർ റാമിന്റെ രൂപകല്പന. നിർദ്ദേശങ്ങളുടെ പട്ടിക CPU വിന്റെ ഇൻസ്ട്രക്ഷൻ സെറ്റായി. സീപിയുവിനു തന്നെ അതിലെ അവസ്ഥകൾ മാറാൻ കഴിയുന്നു. ഏത് ട്യൂറിങ്ങ് മെഷീനാണോ ഈ യൂണിവേഴ്സിറ്റി ട്യൂറിങ്ങ് മെഷീനിൽ സിമുലേറ്റ് ചെയ്യേണ്ടത് അതാണ് പ്രോഗ്രാമിങ്ങ് ഡാറ്റ. റാമിൽ വെയ്ക്കുന്ന ഈ ഡാറ്റ കമ്പ്യൂട്ടർ പ്രവർത്തിപ്പിക്കുന്നു. ഒന്നും പൂജ്യവും അടങ്ങുന്ന നിർദ്ദേശങ്ങളുടെ പട്ടികയില്ല. പകരം മനുഷ്യഭാഷയോടടുത്ത ഹൈ ലെവൽ ഭാഷകൾ. നാട ഇടത്തോട്ടോ വലത്തോട്ടോ ഒരു തവണ നീക്കുന്നതൊക്കെ എത്ര പഴഞ്ചൻ - ഇന്ന് റാമിൽ നിന്നും എവിടെ നിന്നും വേണമെങ്കിൽ ഒറ്റയടിയ്ക്കു വായിക്കാം - Random Access Memory. ഒരു തവണ ഒരു അക്ഷരമാണോ വായിക്കുന്നത്? അല്ലേയല്ല !.

എങ്കിലും...

ഇന്നത്തെ എത്ര ആധുനിക കമ്പ്യൂട്ടറും ട്യൂറിങ്ങ് മെഷീനേക്കാൾ മികച്ചതല്ല. വേഗത കൂടിയതുമത്രമാണ്. ട്യൂറിങ്ങ് മെഷീൻ ചെയ്യാൻ കഴിയാത്തതൊന്നും ഒരു കമ്പ്യൂട്ടറിനും ചെയ്യാൻ കഴിയില്ല!

ട്യൂറിങ്ങ് പരിശോധന

കാലം കറേ കഴിയുമ്പോൾ മനുഷ്യന്റെ അത്രയും ബുദ്ധിയുള്ള ഒരു യന്ത്രം കണ്ടുപിടിക്കപ്പെടുവെന്നിരിക്കട്ടെ. അതൊരു റോബോട്ടോ, അല്ലെങ്കിൽ ഒരു സോഫ്റ്റ്‌വെയർ പ്രോഗ്രാമോ എന്തെങ്കിലുമാവട്ടെ. എങ്ങനെ നമുക്കു പറയാനാവും ആ മനുഷ്യബുദ്ധി പ്രകടിപ്പിക്കുന്നുവെന്ന്? ഇതിനായി അലൻ ട്യൂറിങ്ങ് നിർദ്ദേശിച്ച ഒരു പരിശോധനയാണ് ട്യൂറിങ്ങ് പരിശോധന (Turing Test). ലളിതമായി പറഞ്ഞാൽ ആ പരിശോധന ഇപ്രകാരമാണ്: ഒരു റോബോട്ടാണ് ഈ അവകാശവാദവുമായി നമുക്കു മുന്നിലെന്നിരിക്കട്ടെ. ഒരു മനുഷ്യനെയും ഈ റോബോട്ടിനെയും രണ്ട് മുറികളേക്ക് കയറ്റി വാതിലടയ്ക്കുക. റോബോട്ട് ഏതുമുറിയിൽ, മനുഷ്യനേതു മുറിയിൽ എന്ന് അറിയാത്ത ഒരു വിധികർത്താവും നമുക്ക് വേണം. ഇദ്ദേഹത്തിന് രണ്ടു് പേരോടും കുറേ ചോദ്യങ്ങൾ ചോദിക്കാം. ചോദ്യങ്ങൾക്കുള്ള ഉത്തരങ്ങളിൽ നിന്ന് ഏതുമുറിയിലാണ് റോബോട്ട് ഏതുമുറിയിലാണ് മനുഷ്യൻ എന്നു പറയുകയാണ് വേണ്ടത്. ഈ വിധികർത്താവിന് ഒരു ടെലിഫോൺ ഉപകരണത്തിലൂടെ, അല്ലെങ്കിൽ സംസാരിക്കാത്ത ആളുടെ വ്യക്തിത്വം മനസ്സിലാക്കാൻ പറ്റാത്ത രീതിയിൽ (സംസാരിക്കാൻ പറ്റില്ല, കാരണം ശബ്ദം കേട്ടു് തിരിച്ചറിഞ്ഞാലോ) ആയിരിക്കണം ചോദ്യങ്ങൾ ചോദിക്കേണ്ടത്. റോബോട്ടും മനുഷ്യനും ഉത്തരങ്ങൾ കൊടുക്കേണ്ടതും ഇപ്രകാരം തന്നെ. കണ്ടുപിടിക്കാൻ കഴിയുമോ കിട്ടുന്ന ഉത്തരങ്ങളിൽ നിന്ന് ആരാ യന്ത്രമെന്ന്? ബുദ്ധിപൂർവ്വവും ശരിയുമായ ഉത്തരങ്ങൾ റോബോട്ട് തരികയാണെങ്കിൽ ഈ വിധികർത്താവ് അത് ഒരു മനുഷ്യനാണെന്ന് പറയാനിടവരില്ല. 2000ത്തോടു കൂടി വിധികർത്താവ് ഇങ്ങനെ റോബോട്ടിനെ മനുഷ്യനാണെന്നു കരുതാനുള്ള സാധ്യത 20%ത്തോളം ആകുമെന്ന് പ്രവചിച്ചു ട്യൂറിങ്ങ്. ആർട്ടിഫിഷ്യൽ ഇൻ്റലിജൻസ് ഈ പ്രവചനത്തെ സാധൂകരിക്കാനുള്ള നിലയിലെത്തിയോ ഈ 2012 ലും?



ചിത്രം 4: ട്യൂറിങ്ങ് ടെസ്റ്റിന്റെ ചിത്രീകരണം. C എന്നയാൾക്ക് A, B എന്നിവരോടു ചോദ്യങ്ങൾ ചോദിച്ച് അതിൽ ഏതാണ് മനുഷ്യൻ അല്ലെങ്കിൽ മെഷീൻ എന്നു പറയാൻ സാധിക്കുന്നു. പറയാൻ സാധിച്ചില്ലെങ്കിൽ മെഷീൻ ട്യൂറിങ്ങ് പരിശോധനയിൽ വിജയിച്ചിരിക്കുന്നുവെന്ന് പറയാം.

എനിഗ്മ

രണ്ടാം ലോകമഹായുദ്ധകാലത്ത് ജർമ്മനി ഉപയോഗിച്ചിരുന്ന സന്ദേശങ്ങളെ രഹസ്യസന്ദേശങ്ങളാക്കി മാറ്റാനുള്ള യന്ത്രമായിരുന്നു എനിഗ്മ മെഷീൻ. വളരെ ലളിതമായി പറഞ്ഞാൽ സന്ദേശത്തിലെ അക്ഷരങ്ങളെയെല്ലാം ഒരു പ്രത്യേക സൂത്രം ഉപയോഗിച്ച് മാറ്റിമറിയ്ക്കാനുള്ള യന്ത്രം. ശതകളുടെ കയ്യിൽ ഈ സന്ദേശങ്ങൾ കിട്ടിയിട്ടൊരു കാര്യവുമില്ല, തിരിച്ച് ഈ അക്ഷരങ്ങളെ എങ്ങനെ മാറ്റിമറിച്ച് ശരിക്കുമുള്ള സന്ദേശമാക്കാൻ കഴിയും എന്നറിയേണ്ട? അത് എനിഗ്മ മെഷീനുമത്രമേ കഴിഞ്ഞിരുന്നുള്ളൂ. യുദ്ധക്കപ്പലുകളുടെ ആക്രമണലക്ഷ്യങ്ങൾ എനിഗ്മ മെഷീന്റെ സന്ദേശങ്ങൾ കൊണ്ട് നിയന്ത്രിച്ചിരുന്നതിനാൽ ജർമ്മനിയുടെ ശത്രുരാജ്യങ്ങളുടെ രഹസ്യാനുഷ്ഠാന ഉദ്യോഗസ്ഥർ ശരിക്കും വലഞ്ഞു.



ചിത്രം 5: രണ്ടാം ലോകമഹായുദ്ധകാലത്തെ എനിഗ്മ മെഷീൻ.

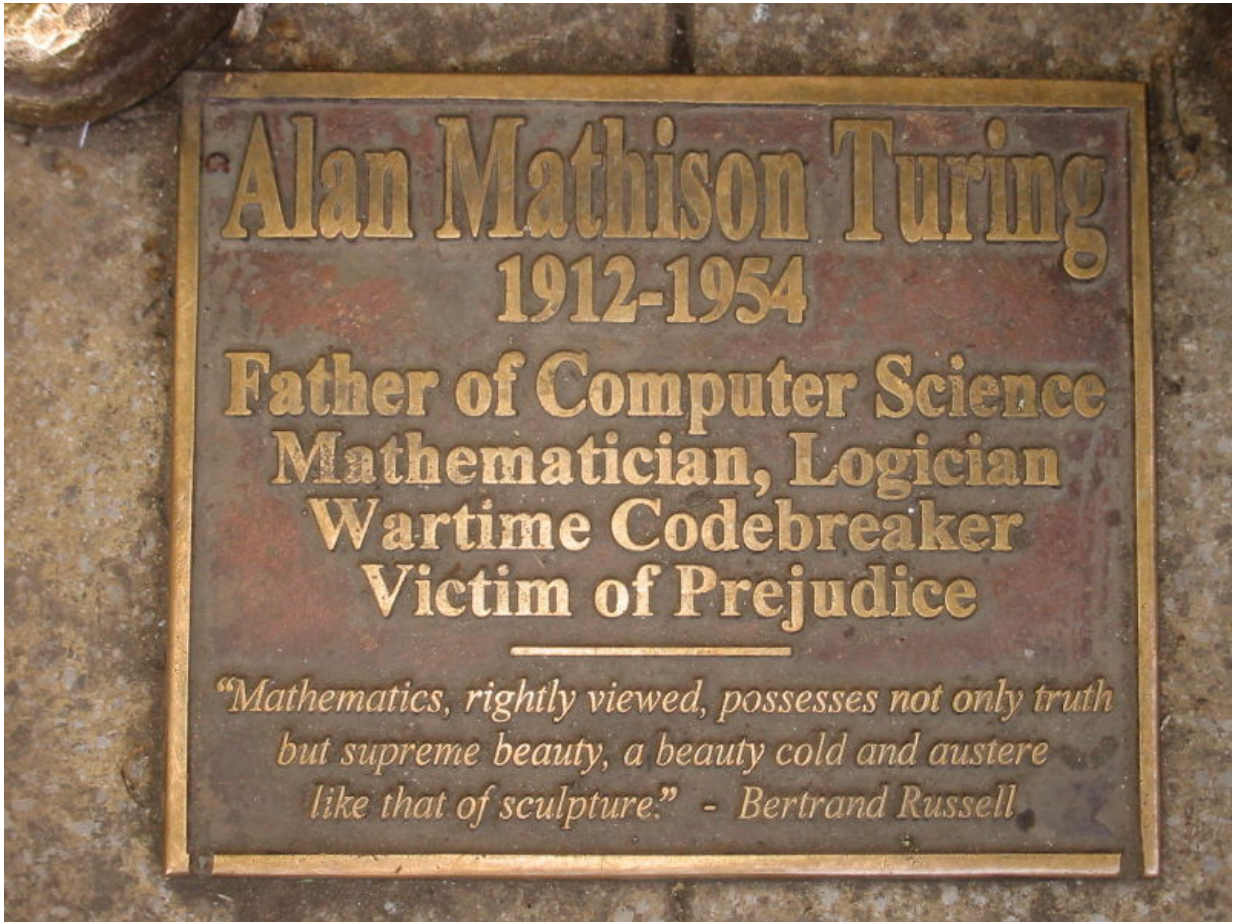
ബ്രിട്ടീഷ് നിർദ്ദേശപ്രകാരം 1939ൽ ബ്ലൈഷ്ലി പാർക്കിലെ രഹസ്യസന്ദേശങ്ങൾ കൈകാര്യം ചെയ്യുന്ന കേന്ദ്രത്തിൽ യുദ്ധകാലസേവനത്തിനെത്തി. അക്കാലത്ത് പോളിഷ് സൈന്യം എനിഗ്മ കോഡുകൾ പൊളിക്കാൻ ബോമ്പ് (Bombe) എന്ന ഒരുപകരണം ഉപയോഗിക്കുന്നുണ്ടായിരുന്നു. അത് ബ്രിട്ടീഷ് സൈന്യത്തിനും അറിവുള്ളതായിരുന്നു. പക്ഷേ ഇത് ഒട്ടും കുറ്റമറ്റതായിരുന്നില്ല. അലൻ ട്യൂറിങ്ങ് നടത്തിയ പരീക്ഷണങ്ങളുടെ ഫലമായി വളരെ വിശ്വസനീയമായ ഒരു പുതിയ ബോമ്പ് മെഷീൻ നിർമ്മിയ്ക്കാൻ ബ്രിട്ടനു കഴിഞ്ഞു. ഇതിനെത്തുടർന്ന് അമേരിക്കക്കു വേണ്ടി എനിഗ്മ കോഡ് പൊളിക്കാനും ബോമ്പ് മെഷീൻ ഉണ്ടാക്കാനും 1942 ൽ ട്യൂറിങ്ങ് അമേരിക്ക സന്ദർശിച്ചിരുന്നു.



ജോലി ചെയ്തിരുന്ന ബ്ലൈക്ക്ലി പാർക്കിൽ നിന്ന് ലണ്ടനിലേക്ക് 60 കിലോമീറ്റർ ഓടിയ ചരിത്രമുണ്ട് ട്യൂറിങ്ങിന്. 1948 ൽ ബ്രിട്ടൻ ഒളിമ്പിക്സിന് ആതിഥേയത്വം വഹിച്ചപ്പോൾ ബ്രിട്ടീഷ് മാർത്തോൺ ടീമിനു വേണ്ടി നടത്തിയ പരീക്ഷണഓട്ടത്തിൽ ട്യൂറിങ്ങ് അഞ്ചാമതെത്തി. 2012 ൽ വീണ്ടും ഒളിമ്പിക്സ് ലണ്ടനിലെത്തുന്നു. ഇപ്രാവശ്യത്തെ മാർത്തോൺ മത്സരത്തിന് ട്യൂറിങ്ങ് മാർത്തോൺ എന്ന പേരിടണമെന്ന് ഒരു ക്യാമ്പയിൻ നടക്കുന്നുണ്ട്.

മരണം

ശാസ്ത്രലോകത്തെ ഈ മഹാനായ പ്രതിഭയ്ക്ക് പക്ഷേ ചരിത്രം വിധിച്ചത് ക്രൂരമായ അവസാനമായിരുന്നു. സ്വവർഗ്ഗ-ലൈംഗിക കറ്റകരമായ നാളുകളായിരുന്നത് അത്. അലൻ ട്യൂറിങ്ങിന്റെ സ്വവർഗ്ഗലൈംഗികത തിരിച്ചറിഞ്ഞ ബ്രിട്ടീഷ് പോലീസ് അദ്ദേഹത്തെ 1952 മാർച്ച് 31 ന് അറസ്റ്റ് ചെയ്തു. തന്റെ ലൈംഗികത തുറന്നു പറയുന്നതിൽ ഒരു തെറ്റും ട്യൂറിങ്ങ് കണ്ടിരുന്നില്ല. ജയിലിലേക്ക് പോകുന്നതിനുപകരം ഹോർമോൺ ചികിത്സ ട്യൂറിങ്ങ് സ്വീകരിച്ചു. തന്റെ പരീക്ഷണങ്ങൾ ട്യൂറിങ്ങ് തുടർന്നു. മോർഫോജനറ്റിക് മേഖലയിൽ അദ്ദേഹം പല പഠനങ്ങളും നടത്തി. ഇലകളിലും സൂര്യകാന്തിച്ചിടിയിലും ഒക്കെ കാണുന്ന വലയങ്ങളും ഫിബൊനാച്ചി ശ്രേണിയും തമ്മിലുള്ള ബന്ധത്തെക്കുറിച്ച് അദ്ദേഹം പഠിക്കാനാരംഭിച്ചിരുന്നു. പക്ഷേ 1954 ജൂൺ 4 ന് അദ്ദേഹം സയനൈഡ് ഉള്ളിൽ ചെന്ന് മരിച്ച നിലയിൽ കാണപ്പെട്ടു. പാതി ഭക്ഷിച്ച ഒരു ആപ്പിൾ മൃതദേഹത്തിനടുത്തുണ്ടായിരുന്നു. അദ്ദേഹത്തിന്റെ അമ്മ ഒരു രസതന്ത്രപരീക്ഷണത്തിൽ അബദ്ധത്തിൽ സയനൈഡ് ഉള്ളിൽ ചെന്നതാണ് മരണത്തിനു കാരണം എന്നു വിശ്വസിച്ചു. പക്ഷേ ആപ്പിളിൽ സയനൈഡിന്റെ അംശം കണ്ടെത്താനായില്ല.



ചിത്രം: മാഞ്ചസ്റ്ററിലെ സാക്ക്വില്ലി പാർക്കിലെ ട്യൂറിങ്ങ് സ്മാരക ഫലകം.

2009ൽ ട്യൂറിങ്ങിന്റെ ആരാധകരായ ശാസ്ത്രപ്രേമികൾ ബ്രിട്ടീഷ് സർക്കാർ ട്യൂറിങ്ങിനെ സ്ഥവർഗ്ഗലൈംഗികതയ്ക്ക് ശിക്ഷിച്ചതിന് മാപ്പു പറയണമെന്ന ആവശ്യം ഉന്നയിക്കുകയും ആയിരക്കണക്കിനാളുകൾ ഒപ്പിട്ട ഒരു ഭീമഹർജി തയ്യാറാക്കുകയും ചെയ്തു. ഇതിനെത്തുടർന്ന് അന്നത്തെ ബ്രിട്ടീഷ് പ്രധാനമന്ത്രി ഗോഡൻ ബ്രൗൺ 2009 സെപ്റ്റംബർ പത്തിന് ലോകത്തോടായി ഇങ്ങനെ പ്രഖ്യാപിച്ചു.

"അലൻ ട്യൂറിങ്ങിന് നീതി ആവശ്യപ്പെട്ടും അദ്ദേഹത്തിനുണ്ടായ ശ്രോണഭവങ്ങൾക്ക് മാപ്പ് ആവശ്യപ്പെട്ടും ആയിരക്കണക്കിനാളുകൾ മുന്നോട്ടുവന്നിരിക്കുന്നു. അന്നത്തെ നിയമപ്രകാരമായിരുന്നു ട്യൂറിങ്ങിനെ ശിക്ഷിച്ചതെങ്കിലും നമുക്ക് പഴയകാലത്തേയ്ക്ക് പോകാനാകില്ല. തീർച്ചയായും അദ്ദേഹത്തിനുണ്ടായ ദുരനുഭവങ്ങളിൽ ഞാൻ അഗാധമായി വേദിക്കുന്നു. ബ്രിട്ടീഷ് ഗവൺമെന്റിനുവേണ്ടി ഞാൻ പറയുന്നു. ഞങ്ങൾ വേദിക്കുന്നു. ട്യൂറിങ്ങ്, നിങ്ങളോടു ഞങ്ങൾക്ക് ഇതിനേക്കാൾ നന്നായി പെരുമാറാമായിരുന്നു."

ചിത്രങ്ങൾക്കും വിവരങ്ങൾക്കും കടപ്പാട്: വിക്കിപീഡിയ, Alan Turing: a short biography by Andrew Hodges.

ഈ ലേഖനം ക്രിയേറ്റീവ് കോമൺസ് ആട്രിബ്യൂഷൻ ഷെയർ എലൈക് 2.5 ഇന്ത്യ (CC-BY-SA 2.5 In) പ്രകാരം പുനഃപ്രസിദ്ധീകരിക്കാൻ അനുമതി നൽകുന്നു.

[Santhosh Thottingal](#) santhosh.thottingal@gmail.com 2012 June 5